



UNIVERSIDAD NACIONAL AUTÓNOMA  
DE MÉXICO

FACULTAD DE INGENIERÍA

SISTEMA DE ADMINISTRACIÓN Y OPERACIÓN DE  
PROYECTOS PARA EMPRESAS DE CONSULTORÍA DE  
INFORMÁTICA

T E S I S

QUE PARA OBTENER EL TÍTULO DE

INGENIERO EN COMPUTACIÓN

P R E S E N T A N :

GAMALIEL ENRIQUE ALEJO ORDAZ  
ALEJANDRO BESNE PEÑA  
MARICARMEN HERNÁNDEZ CERVANTES  
FORTINO JIMÉNEZ JIMÉNEZ  
SUSANA VALDES MÁRQUEZ

DIRECTOR DE TESIS: M. en I. JUAN CARLOS ROA BEIZA

MÉXICO, D. F.

2000



284327



Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

*A mi esposa Alma Cabrera, y mis hijos, Lizbeth y Leonardo, por que ellos son el motivo de mi constante superación, los amo.*

*A mi madre Teresa Peña, por su valioso apoyo al inicio de mi carrera, la respeto y quiero mucho.*

*Con todo respeto a mi padre Salvador Besné, y hermanos Octavio, Olivia, Salvador, Cecilia y David, los quiero.*

*A mis sobrinos Andrea, Jessica, Sandra, Jesús y Arantxa, por que me gustaría y sería un orgullo para mi verlos recibirse algún día.*

*A mi querida UNAM y Facultad de Ingeniería por su gran y valioso apoyo para mi preparación académica y personal, lo que me ha abierto las puertas en el mundo laboral, gracias.*

*A mis amigos y enemigos por que tambien de ellos se aprende.*

*Alejandro Besné Peña*

Quiero agradecer a tantas personas, que no sé por donde empezar, así que antes de hacerlo, quiero dejar en claro que todas las personas cercanas a mí son muy importantes, ya que cada quien me ha dejado distintas enseñanzas. ¡Gracias!

A mi mamá a quien amo y siento como una verdadera amiga y de quien saque el carácter tenaz; a mis hermanos Saúl y Angel que junto con Aura son mis pequeñitos consentidos; a Mayté por ser una de mis tantas mamás y a quien admiro por su nobleza y el apoyo que siempre me ha dado, a mi abuelita que no me he podido separar de ti nunca aunque lo haya intentado, te quiero mucho; a Ernesto otra de mis "mamás" en quien he encontrado un apoyo muy fuerte y el punto de vista masculino en mi vida, a mi tía Gloria que tanto se preocupa por mí y que siento como me quiere... es mutuo el sentimiento; a mi abuelito que fuiste mi padre también y aunque no estás presente, sigues con nosotros y estoy segura que estarías muy orgulloso.

Socorro mi "novia" que sabes tooda mi vida, en ti he encontrado siempre un apoyo excepcional en todo momento, te quiero muchísimo! y estoy muy orgullosa de ti, ahora si estoy segura de que podemos superar cualquier cosa (juntas, eh?); Fabiola me has enseñado a conocerme más y te agradezco mucho que me presentaras un camino diferente por el cual se es más feliz, te admiro por ser tan buena mamá, esposa, pero sobre todo mujer; Carlitos S (T d l Cielos) gracias por ayudarme a que esto sea ya una realidad, ¡te quiero mucho!; primito Fernando es un gusto que después de tantos años de conocernos haya encontrado una amistad contigo, además te tengo mucha confianza y cariño; Pavel eres un amigo atrabancado y terco, pero de las personas más nobles en el planeta; Carolina G. soportaste junto conmigo el peso de la facultad y salimos adelante! ; Mercedes B. niña inteligente, no que no te casabas? =) te quiero un buen!; Rubén R. te aprendí a fijar metas, pero sobre todo a alcanzarlas! (gracias por las clases de manejo, sé cuanto las sufriste! jeje); Rubén Ch un aliciente por el que seguí en ingeniería, eres muy buen amigo (fresa); Javier M admiro la libertad que tienes y me gusta como me la transmites, además a tu familia también la quiero mucho; Miguel Angel mi papá virtual y muy buen amigo desde hace años, te perdono que te hayas casado, jeje (ya no te cortes); Carlos Torres te admiro por todo lo que sabes, cuanto lees, el buen amigo que eres y cuanto disfrutas la vida; Joaquin hombre casi perfecto que ¡ni pareces mexicano! me encanta tu tenacidad y lo que me transmites, además que tienes muy buen sentido del humor; Pablo *sueño utópico pero a la vez maravillosa experiencia que no cambiaría jamás, ¡eres muy especial!*; Eduardo (maloso topo) eres un niño grandote, pero eso mismo te hace una persona super fresca; Arturo Velásquez que confiaste en mí y me diste una primera oportunidad laboral interesante, de grande quiero ser como tu! =); José Luis siempre tan calladito has marcado todo un estilo! =); Antonio I en lo profesional eres ejemplo a seguir; Gerardo S. fan del nintendo.. ¡te encuentro hasta en EU! gracias por tu amistad y por demostrar que se puede salir adelante a pesar de los obstáculos que nos encontremos; Federico (boro) mi buen amigo guru de las Mac's! aprecio que siempre te preocupes por mí, estoy segura de que pronto estarás en las mismas...; Miguel Eduardo el mejor instructor de negocios que puedo tener a mi lado; Guillermo Deister eres el mejor jefe del mundo, todos te recordamos mucho y me gustaría algún día ser así con la gente que trabaje; Karla R mi querida nena pelirruichis eres super divertida! (no cambies, vales mil jeje); María eres la muestra de que a todos nos llega nuestro príncipe azul!, y de que inteligencia y belleza no

están peleadas; Berenice amiguita más cercana desde que nos fuimos a Cuba, eres una excelente compañera de viaje y amiga; Arturito V que nos das envidia por tu número de conquistas!.. ¿de que te quejas?; Frankie la vocecita de la razón dentro de mi cabecita loca (bueno, mas bien de todos los labastichicos); John Fleites gracias a ti y a Candy por su hospitalidad en Cuba y por enseñarme tu manera de amar ese país, tienes una calidez muy especial; Walter J. wallyamigo tan reventado, eres master en creatividad y un verdadero dulce; Francisco P. super dedicado, tierno y preocupado por mi bienestar, de verdad eres un amigo muy valioso; Rafael M. gracias por tu amistad y porque siempre me has apoyado y empujado a este momento; Zuleika amiga estás super loca... ¿por qué será que me identifico tanto contigo? ¡jaja! sé que vas a ser muy feliz junto con Douglas; Vanessa eres un buen legado que me dejó Zuly (junto con tus primas Liz y Miros), espero que convivamos mucho más!; Pilar E. amiga chiquita de actuación y compañerita de reven, eres un ser precioso por tu calidez; Elika O. más chiquita pero super alivianada y buena amiga; Naye con tu sentido del humor nos haces disfrutarte mucho; Cesar siempre tan callado pero con un gran corazón; Carlos C. la tecnología hizo que nos conociéramos, eres de los amigos más sensibles que tengo, ¡no pierdas eso!; Pedro R tuviste las ganas de salir adelante para bien tuyo y de tu alrededor; Cesar Val España intentando reconquistar a México!; Abdelkrim quien me aclaró mis dudas acerca de los musulmanes y eres tan detallista y tierno!; Rafael S eres el puertorriqueño en quien tengo puesta una esperanza, me gusta como has planeado tu vida y lo reventado que eres.

Juan Carlos Roa sin ti el camino tal vez lo habríamos recorrido, pero estoy segura de que nos lo aligeraste mucho con tu experiencia y el apoyo que nos diste ¡Gracias!; a mis sinodales que apoyan la realización de este sueño, a mi equipo de tesis que no por últimos son menos importantes, espero que este logro se multiplique por muchos más en su vida profesional!

Si omití a alguien, seguro es problema de memoria nada mas, no falta de cariño. A todos mi amor porque cada uno ha dejado un poco de sí mismo en mí.

Mary

# Dedicatoria

Este trabajo esta dedicado especialmente a mis padres por el apoyo incondicional:

A mi papá por darme la responsabilidad y el carácter de ser.

A mi mamá por los cuidados y valores inculcados.  
Gracias, los quiero mucho.

A mi hermano y mi amigo Eusebio, por los tiempos buenos y malos que hemos pasado juntos.

A mis hermanas Margarita y Gela por su cariño y por soportar mis malos ratos.

A Raúl †  
Aún te recuerdo por tu gran amistad.  
Q.E.P.D.

A Marcela †  
Amiga, este trabajo esta dedicado a tu memoria, por la tesis que no terminamos juntos. Quiero agradecerte habernos conocido. Gracias.  
Q.E.P.D.

A Francisco Cortés:  
El mejor jefe que he tenido. Gracias por tu apoyo.

A mis amigos:  
Cuando Dios creó a los amigos, lo hizo pensando en que tuviéramos la posibilidad de elegir a un ser querido en el cual pudiéramos confiar.  
Gracias por su amistad.

Fortino Jimenez J.

AGRADECIEDO A MIS PADRES:  
JERÓNIMO VALDÉS GONZALEZ  
GUADALUPE MÁRQUEZ HERNÁNDEZ  
POR BRINDARME EL APOYO ECONÓMICO Y MORAL PARA  
PODER CONCLUIR MIS ESTUDIOS

*A todos mis hermanos:*

*María, Mario, Carlos, Aida, Jerónimo y Ana*

*En especial a mi hermano Mario que con su ejemplo de estudio y orientación,  
he estudiado y concluido esta carrera.*

*A mis hijas y compañero:*

*Syumar, Kinyao y Marcelino, porque representan mi fuerza para seguir  
adelante.*

*A Doña Fere:*

*Por su siempre apoyo incondicional*

*A mis compañeros:*

*Por haber realizado el trabajo en equipo para que se diera este trabajo.*

*Gracias a todos*

*Susana Valdés*

<b>Capítulo 1 Reglas del Negocio</b>	<b>1</b>
<b>1.1 ENTORNO DE UNA EMPRESA CONSULTORA</b>	<b>1</b>
1.1.1 Giro de la empresa	1
1.1.2 Organización de la empresa	3
1.1.3 Funciones de las áreas	4
1.1.4 Estrategia del servicio	7
<b>1.2 CONCEPTOS BÁSICOS DE ADMINISTRACIÓN DE PROYECTOS</b>	<b>12</b>
1.2.1 ¿Qué es el proyecto?	12
1.2.2 Elementos que constituyen un proyecto	12
1.2.3 El medio ambiente del proyecto.	15
1.2.4 Tipos de proyectos	31
1.2.5 Clasificación de proyectos	31
1.2.6 Fase organizacional	31
1.2.7 Fase de construcción	34
<b>Capítulo 2 Teoría Básica</b>	<b>45</b>
<b>2.1 FUNDAMENTOS DE LA PROGRAMACIÓN ESTRUCTURADA</b>	<b>45</b>
<b>2.2 METODOLOGÍA DE DISEÑO Y ANÁLISIS DE SISTEMAS</b>	<b>57</b>
<b>2.3 SISTEMAS OPERATIVOS</b>	<b>72</b>
2.3.1 ¿Qué es un Sistema Operativo?	73
2.3.2 Interfaz de Línea de Comandos	75
2.3.3 Interfaz Gráfica del Usuario	75
2.3.4 Funciones de los Sistemas Operativos	76
2.3.5 Sistema Operativo Multitareas	76
2.3.6 Sistema Operativo Monotareas	77
2.3.7 Sistema Operativo Monousuario	78
2.3.8 Sistema Operativo Multiusuario	78
2.3.9 Secuencia por Lotes	79
2.3.10 Tiempo Real	79
2.3.11 Tiempo Compartido	80
2.3.12 Los Sistemas Operativos más Populares de las PC	80
2.3.12.1 OS/2	80
2.3.12.2 Macintosh OS	81
2.3.12.3 UNIX	83
2.3.12.4 Microsoft Windows NT	84
2.3.12.5 Características	84
2.3.12.6 Requerimientos de hardware	85
2.3.12.7 Servicios	86
2.3.12.8 Configuraciones de discos	91
2.3.12.8.1 NTFS	92
2.3.12.9 FAT	92
2.3.12.10 HPFS/HPFS386	93
2.3.12.11 Configuración de dominios, restricciones de usuarios y grupos	93
2.3.12.12 Manejo de replicasiones de directorios	95
2.3.12.13 Manejo de la seguridad	96
2.3.12.14 Servicios de impresión	97
2.3.12.15 Memoria virtual	97
2.3.12.16 Protocolos	97
2.3.12.5 Microsoft Windows 95	100
2.3.12.6 LINUX	101
2.3.12.7 Qué beneficios ofrece Linux.	102

2.3.12.8 Fragmentación de Archivos.	103
2.3.12.9 Administración de Memoria.	104
<b>2.4 COMUNICACIONES</b>	<b>105</b>
2.4.1 Medios, Formas Y Tipos De Transmisión	107
2.4.2 Protocolos	109
2.4.3 Redes WAN	115
2.4.4 Redes De Area Local	123
<b>2.5 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DE POWERBUILDER 5.0</b>	<b>128</b>
2.5.1 Versiones De Powerbuilder	128
2.5.2 Qué Es Powerbuilder	129
2.5.3 El Ambiente De Powerbuilder	130
2.5.4 Arquitectura Cliente / Servidor.	133
2.5.5 Aplicaciones Desarrolladas Con Powerbuilder	134
2.5.6 Ayuda De Línea Y Documentación	135
2.5.7 Ambiente De Diseño Y Desarrollo De Power Builder 5.0	135
2.5.8 Definición De Powerscript Lenguaje	142
2.5.9 Características Técnicas De Powerbuilder	156
<b>2.6 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DE SQL ANYWHERE</b>	<b>158</b>
<b>Capítulo 3 Planteamiento del Problema y Propuesta de Solución</b>	<b>163</b>
<b>3.1 PLANTEAMIENTO DEL PROBLEMA</b>	<b>163</b>
3.1.1 Situación Actual	163
3.1.2 El Usuario y sus Requerimientos	173
3.1.3 Búsqueda y Análisis de la Información	177
3.1.4 Identificación del Problema	184
3.1.5 Áreas Involucradas en el Proceso	189
<b>3.2 OPCIONES DE SOLUCIÓN</b>	<b>192</b>
3.2.1 Elección de la Solución Óptima	195
3.2.2 Herramientas de Desarrollo	202
<b>Capítulo 4 Desarrollo e Implementación del Sistema</b>	<b>208</b>
<b>4.1 APLICACIÓN DE LA METODOLOGÍA ELEGIDA (Back end para cada módulo)</b>	<b>208</b>
4.1.1 Diagrama de Contexto	208
4.1.2 Diagrama de Flujo de Datos	211
4.1.3 Diccionario De Datos	234
4.1.4 Diagrama Entidad-Relación	247
4.1.5 Normalización	254
<b>4.2 CONSTRUCCIÓN DEL BACK END.</b>	<b>267</b>
4.2.1 Generación De La Base De Datos Y Sus Elementos	268
4.2.2 Construcción De Tablas	274
4.2.3 Llaves Foráneas (Forcing Key)	279
4.2.4 Índices (Index)	284
4.2.5 Desencadenadores (Trigger)	287
4.2.6 Código Generado Para El Esquema De Bases De Datos	292
<b>4.3 DISEÑO Y CONSTRUCCIÓN DEL FRONT-END</b>	<b>335</b>
4.3.1 Ambiente De Diseño Y Desarrollo De Power Builder 5.0	336
4.3.2 Diseño Y Construcción De Un Proyecto	355
<b>4. 4 PRUEBAS E INTEGRACIÓN DEL SISTEMA</b>	<b>395</b>

4.4.1 Pruebas del Software	395
4.4.2 Estrategias de Prueba	396
4.4.3 Tipos de Pruebas	397
4.4.4 Modulo "Proyectos"	398
4.4.5 Modulo "Evaluación del Proyecto"	404
4.4.6 Pruebas de Recuperación	406
4.4.7 Pruebas de Desempeño	407
4.4.8 Transición	407
4.4.9 Capacitación a los Usuarios	407
4.4.10 Pruebas en Paralelo	408
<b>4.5 GENERACIÓN DE REPORTES PARA LA TOMA DE DECISIONES</b>	<b>409</b>
<b>4.6. FACTIBILIDAD TÉCNICA Y OPERATIVA</b>	<b>434</b>
4.6.1 Requerimientos de Hardware	435
4.6.2 Requerimientos de Software	436
4.6.3 Requerimientos de Información	436
4.6.4 Implementación	437
4.6.5 Costo del Sistema	437
4.6.6 Plan de Desarrollo	439
4.6.7 Capacitación	440
<b>Manual Técnico</b>	<b>442</b>
<b>Manual de Usuario</b>	<b>450</b>
<b>Conclusiones</b>	<b>532</b>
<b>Bibliografía</b>	<b>535</b>
<b>Apéndice</b>	<b>536</b>

# Capítulo 1

## Reglas del Negocio

### 1.1 ENTORNO DE UNA EMPRESA CONSULTORA

#### 1.1.1 Giro de la empresa

Para cualquier empresa consultora la capacitación constante del equipo es cuidadosamente estudiada para satisfacer necesidades de la industria con respecto a las diferentes tecnologías y fomentar el desarrollo personal de su gente; con la finalidad de presentar servicios de alto nivel.

El desarrollo de sistemas de información para una variedad de industrias, es quizás, el negocio mas fuerte en una empresa consultora, por ello se requiere no sólo un amplio conocimiento acerca de la tecnología disponible, sino también, un dominio de su aplicación y desarrollo.

Hoy en día al hablar de tecnología disponible, podemos hacer mención de las siguientes Bases de Datos:

IBM DB/2

IBM/DL/I

Informix

Oracle

Sybase

SQL Server

De los siguientes lenguajes:

AS-400 RPG

Informix 4/GL

C, C++

Progress

Centura

Java

Cobol

Power Builder

Visual Basic

HTML

Developer / 2000

Y de los siguientes Sistemas Operativos:

MVS

UNIX

VMS

VSE

Windows

NT

También se ofrece servicios de consultoría en los siguientes campos.

- 1) Telecomunicaciones y planeación estratégica de sistemas de información
- 2) Evaluación e implementación de paquetería de software.
- 3) Reingeniería de procesos de negocio.
- 4) Implementación de procesos de negocio.
- 5) *Plan de negocios.*
- 6) Ajuste del plan estratégico (sistemas de información) al plan de negocios.

La mayor parte de las empresas consultoras ofrece cursos de capacitación que pueden ser impartidos ya sea en las instalaciones del cliente o en las propias instalaciones de

capacitación de la empresa.

Cabe mencionar que algunas consultorías manejan sólo ciertas plataformas y ciertos lenguajes, algunas otras sólo software en línea.

### 1.1.2 Organización de la empresa

La empresa se divide en áreas o departamentos, los cuales son dependientes unos de otros, en la mayor parte de las consultoras en un inicio se cuenta con un área de operaciones, un área comercial, un área de recursos humanos y un área financiera; cuando la consultora crece, este sistema no es el más conveniente, ya que como el número de proyectos se extiende, es muy complicado que el encargado del área de operaciones pueda estar al tanto de las necesidades de cada proyecto (en las empresas grandes se llega a manejar mas de 500 proyectos a la vez y en el área de operaciones la cantidad de personal llega a sobrepasar las 1000 personas).

Por eso es necesario que el área de operaciones se divida en áreas más pequeñas que por lo regular son llamadas divisiones; cada división tendrá a su cargo un número de proyectos y una cartera de clientes con sus propios vendedores, y su propio equipo de programadores, líderes de proyecto, consultores, etcétera.

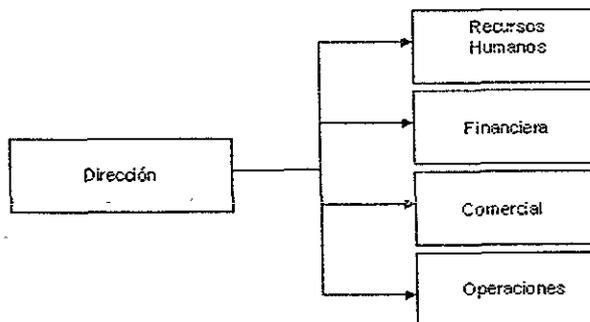


FIGURA. 1.1.2.1 Organización de una empresa consultora pequeña

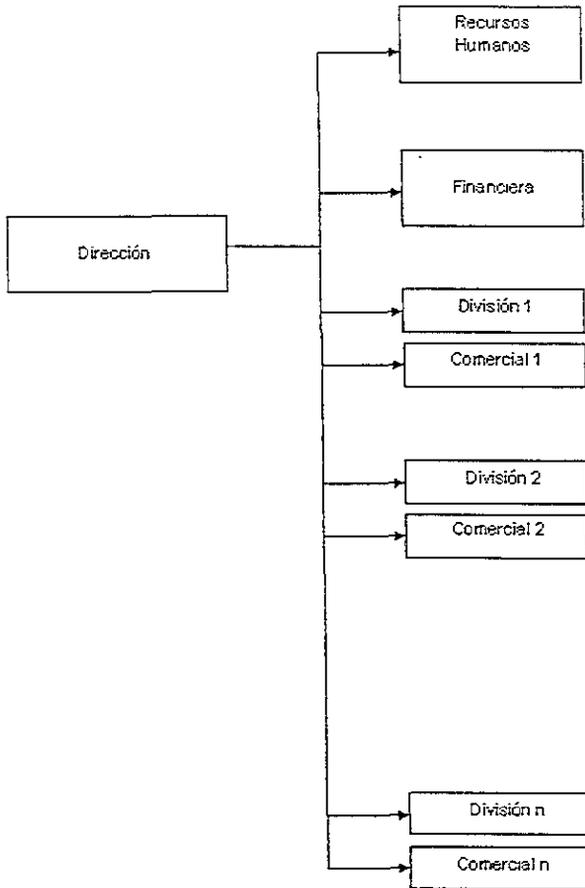


FIGURA. 1.1.2.2 Organización de una empresa consultora grande

### 1.1.3 Funciones de las áreas

Cada área juega un papel importante dentro de la empresa, de hecho para que todo funcione de manera correcta debe haber comunicación y retroalimentación de cada una de las áreas; cabe mencionar que en algunas consultorías ya se han integrado algunos otros departamentos como el de calidad y el de Internet, pero esto no afecta a la estrategia de servicio.

A continuación se describen las funciones principales de cada área.

## Área de Operaciones

Como se había mencionado, las operaciones de la empresa están organizadas en divisiones independientes. Cada división consiste de un Director, ejecutivos de cuenta, líderes de proyecto, ingenieros y analistas para formar un equipo de negocio completamente funcional.

Manteniendo equipos de negocio pequeños, la empresa puede brindar un mejor servicio a sus clientes, y canalizar los problemas de una mejor manera, ahora cada división es responsable de sus triunfos y derrotas, se trata de evitar que una división se involucre con otra, sólo se llegan a involucrar bajo casos muy necesarios, por razones de que sigue siendo una sola empresa.

La función principal de operaciones es el desarrollo de sistemas de información y los servicios de consultoría que se habían mencionado con anterioridad como telecomunicaciones y planeación estratégica de sistemas de información, evaluación e implementación de paquetería de software, reingeniería de procesos de negocio, implementación de procesos de negocio, plan de negocios, ajuste del plan estratégico (sistemas de información) al plan de negocios.

Además, también esta área junto con recursos humanos se encarga de una parte de la capacitación de su personal y de capacitación a personal externo.

## Área de Recursos Humanos

Sus funciones principales son: Integración, capacitación y administración de personal.

La integración de personal contempla el reclutamiento, selección, comunicación externa, inducción, y eventos.

La capacitación contempla desarrollo, estándares, capital intelectual, formatos, estilos,

plan de carrera y becarios.

La administración de personal se refiere a la valuación de puestos, evaluaciones, altas, bajas, vacaciones, IMSS y demás prestaciones dependiendo de la empresa consultora.

### **Área Comercial**

Esta área se encarga de las ventas, de promover los servicios que proporciona la empresa consultora, de presentar y concertar citas con el personal de la empresa según sea necesario de dar información a los clientes acerca de la tecnología y recursos actuales.

Por lo regular después de que un cliente se interesa en alguno de los servicios de la empresa, la primer cita es con uno de los vendedores y tal vez con gente del área de operaciones.

El personal de dicha área se encarga de hacer una propuesta económica de lo que cuesta el servicio que puede ofrecer la consultora, al final el cliente puede negociar dicha propuesta siempre y cuando la consultora esté dispuesta a hacerlo.

Una vez que se acepta la propuesta y durante el transcurso del proyecto el vendedor hace un seguimiento de la parte comercial, hasta que el proyecto es finalizado.

### **Área financiera**

Su función principal es administrar correctamente los recursos financieros de la empresa, desde la valuación de puestos de su personal hasta llevar la contabilidad de la empresa, venta de proyectos, pago de impuestos, nómina, etcétera.

Cabe Mencionar que si la organización de la empresa contempla divisiones en sustitución de una sola área de operaciones, cada división tendrá su propia área

comercial para tener un mejor control.

#### 1.1.4 Estrategia del servicio

Actualmente las empresas de consultoría tienen casi la misma estrategia de servicio, sólo varía en algunos procesos internos pero en esencia se basan en lo que se describe enseguida.

Las empresas fuertes manejan una gran cantidad de proyectos, por lo que la toma de decisiones no siempre es oportuna y/o adecuada.

Cada empresa se ve enfrentada a saber que recursos humanos y materiales pueden utilizar para su asignación a un proyecto específico, cual es el tiempo que tardara cada etapa de un proyecto, si posee los recursos para una de las etapas o un nuevo proyecto, como ha sido el desempeño de los recursos dentro del proyecto, la evaluación del proyecto desde el punto de vista humano y de resultados a nivel general, cuando se podrá disponer de un recurso para ser reasignado, los perfiles de los recursos solicitados por el cliente y si la empresa cuenta con dichos recursos, que proyectos están en espera de recursos, las razones por las que se cancelo un proyecto o no ha sido autorizado, las tecnologías y herramientas requeridas es decir una gran cantidad de información que necesita la empresa, para dar un excelente servicio de consultoría.

Por otra parte las empresas de consultoría en sus inicios contaban con un mínimo de personal, lo cual era perfectamente administrable llevando el control en papel, documentos de texto u hojas de calculo, pero hoy en día esto no es posible debido al crecimiento de estas empresas en personal, tecnología y mercado.

Como lo han venido haciendo las empresas desde hace muchos años, la manera como una consultora comienza a trabajar es de la forma tradicional de cualquier otra empresa, es decir que un vendedor vaya a una empresa y ofrezca los servicios de la compañía consultora o que lo haga por medio telefónico, pero hoy en día se cuenta con

otras formas de anunciarse como Internet, sea cual sea la forma como una empresa se entera de los servicios que ofrece la consultoría, lo primero que se hace es concertar una cita con un vendedor o ejecutivo de cuenta.

Cabe mencionar que cada vendedor maneja una cartera de clientes, y por lo tanto cada división de la empresa tendrá sus propios clientes. Durante la cita se conoce el problema, y se hace una promesa de valuación para una próxima cita.

Junto con personal del área de operaciones se hace una propuesta tanto de tiempo como económica para que después sea revisada junto con el cliente

La información que se maneja para poder llegar a la propuesta, es entre otra: nombre del cliente y el tipo de proyecto, su alcance, el giro, tamaño del proyecto, el tipo de recursos que serán necesarios y la cantidad, el tiempo estimado a cada etapa del proyecto (para un mejor servicio, casi siempre se divide en etapas dicho proyecto), el costo de los recursos tanto humanos como materiales, etc.

Una vez que se tiene la propuesta se trata con el cliente y se llega a un acuerdo, si el proyecto fue aceptado el siguiente paso es para el área de operaciones.

El área de operaciones revisa los requerimientos del cliente y tratara de dar una solución, viendo si cuenta con los recursos que se requieren, realizando planes de trabajo y determinando si es viable el proyecto con lo que posee en ese momento El área tiene que decidir cuantos recursos serán enviados, si existen los perfiles solicitados, que tiempo se le asocia a cada etapa definida, como se puede empezar y cuando.

En todo este proceso operaciones le envía información al área de recursos humanos para cubrir los requerimientos del personal. El área de recursos humanos tiene que *consultar manualmente en archiveros del personal solicitado*, debido a la gran cantidad de empleados es difícil tener un curriculum actualizado de su personal y saber con

quien su cuenta en ese momento (la mayor parte de su personal se encuentra trabajando en un proyecto debido a la gran demanda del mercado).

En el momento en que recursos humanos dispone del personal con el perfil solicitado (total o parcialmente) se envía la información al área de operaciones para que decida junto con el vendedor y el cliente la autorización del proyecto y posteriormente concluir con la firma del contrato.

Cabe mencionar que generalmente el cliente requiere de respuestas rápidas y exige un excelente servicio.

Durante el desarrollo del proyecto se realizan evaluaciones periódicas de cada área del proyecto, evaluando tanto personal, resultados, tiempos, experiencia, desempeño, disposición, puntualidad, recursos materiales etcétera, y se van atendiendo necesidades de todo tipo, las evaluaciones se hacen sobre formatos hechos en papel. Por ejemplo si un empleado estuvo en algún proyecto donde aprendió una herramienta nueva esto se queda registrado sobre hojas de papel, pero no sobre el curriculum del empleado que mantiene la empresa. Es muy importante saber en que estado se encuentra cada proyecto que lleva la empresa, sobre todo cuando hablamos de un manejo de mas de 100 proyectos.

Cabe mencionar que al iniciar un proyecto pueden o no estar cubiertos todos los requerimientos del cliente, además como un proyecto casi siempre se divide en etapas, cada etapa puede necesitar requerimientos específicos, por estas razones y por razón de evaluación del proyecto, el área de operaciones es la que más mantiene contacto con el cliente, pero también dicha área seguirá manteniendo contacto con el área de recursos humanos, para la evaluación de personal o la requisición de personal, aunque también el área comercial representada por el ejecutivo de cuenta mantiene contacto con el cliente durante este transcurso, a causa de la parte financiera.

En resumen la información más importante para el área de operaciones es la siguiente:

1. Registrar los Proyectos de la empresa.
2. Conocer las etapas de cada proyecto y el estado en que se encuentran.
3. Conocer los requerimientos necesarios y faltantes en cada proyecto.
4. Evaluar periódicamente la situación de los proyectos.
5. Registrar problemas y sugerir soluciones para dar seguimiento eficaz a los proyectos.
6. Registrar encuestas en la conclusión de los proyectos para conocer mejor la situación de los mismos.
7. Asignar y controlar los recursos de los diferentes proyectos.
8. Poder contemplar tanto empleados, como candidatos, personal externo e incluso de la gente que esta por liberarse para la asignación de los proyectos.
9. Registrar la experiencia y los niveles de especialización de los recursos.
10. Evaluar el desempeño de los recursos.
11. Contar con un curriculum actualizado de todos los empleados.
12. Contar con historiales de los proyectos de cada cliente y la gente que participó en los mismos.
13. Tener un módulo de estadísticas que muestre de una manera gráfica la situación y las evaluaciones de los proyectos, lo cual nos permita administrar, controlar, dar seguimiento y evaluar mejor los proyectos y los recursos.
14. Mantener información única y disponible para todas las áreas involucradas

Una vez que el proyecto ha terminado, la consultora da una garantía al sistema para cualquier anomalía que pudiera surgir, por esta razón, la consultora tendrá que estar pendiente del cliente para futuros encuentros. Si el servicio convence al cliente lo mas seguro es que si tiene una nueva necesidad de la consultora, le volverá a llamar.

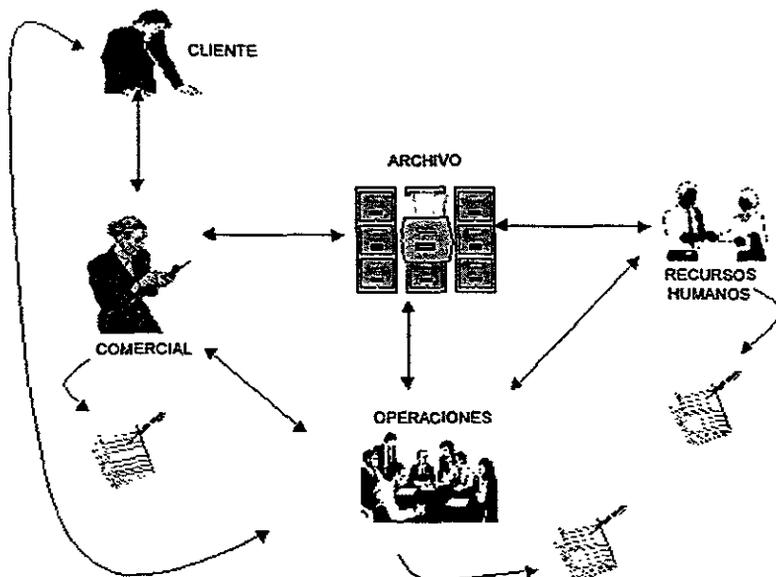


FIGURA. 1.1.4.1 Estrategia del servicio actual

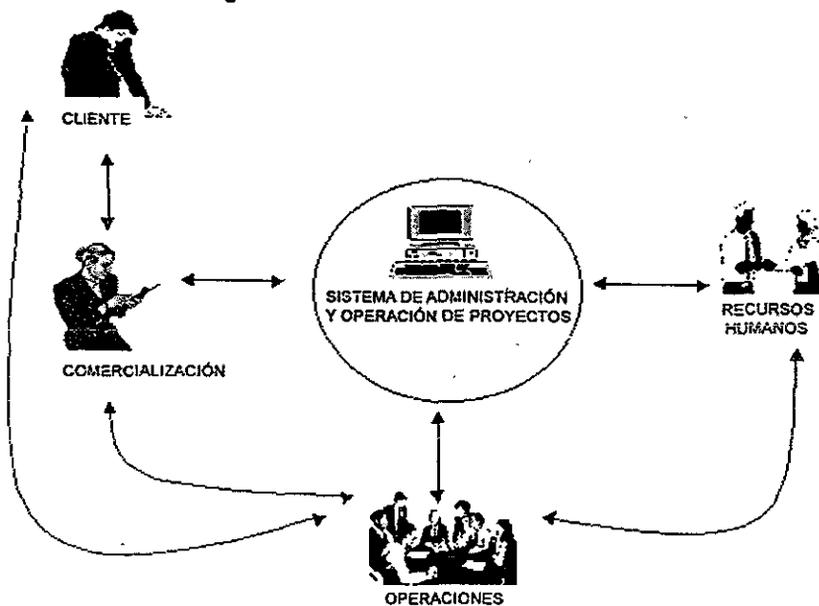


FIGURA. 1:1.4.2 Estrategia del servicio ideal

## 1.2 CONCEPTOS BÁSICOS DE ADMINISTRACIÓN DE PROYECTOS

### 1.2.1 ¿Qué es el proyecto?

El trabajo que desempeñan las organizaciones, generalmente involucra tanto operaciones como proyectos, y estos comparten características similares, como:

- Realizado por gente
- Restricciones por limitación de recursos
- Planeación, ejecución y control.

Las operaciones y proyectos difieren primordialmente en que las operaciones están en progreso y son repetitivos mientras los proyectos son temporales y únicos.

Un proyecto puede estar en términos de una característica distintiva, un proyecto es un esfuerzo temporal para crear un producto o servicio único. Temporalmente porque tiene la intención de tener un inicio y fin definitivo, único porque el servicio o producto es diferente y se distingue de muchos productos o servicios similares.

Los proyectos son emprendidos por todos los niveles de la organización. Y esto puede involucrar a una sola persona o a cientos de ellas. También puede involucrar una o varias áreas de la organización y son muy importantes ya que pueden mejorar las reglas del negocio o de áreas estratégicas.

### 1.2.2 Elementos que constituyen un proyecto

Un proyecto es un intento por lograr un objetivo específico mediante un juego único de tareas interrelacionadas y el uso efectivo de los recursos. Los elementos siguientes ayudan a definir un proyecto:

- Un proyecto tiene un objetivo bien definido, un resultado o producto esperado. Por lo general el objetivo de un proyecto se define en términos de *alcance*, *programa* y *costo*. Por ejemplo, el objetivo de un proyecto pudiera ser introducido al mercado en 10 meses y dentro de un presupuesto de 500,000 pesos un nuevo aparato para la

preparación de alimentos, que cumpla con ciertas especificaciones de desempeño definidas por anticipado. Además, se espera que el alcance del trabajo se logrará con *calidad* y a *satisfacción del cliente*. Un proyecto se lleva a cabo mediante una serie de *tareas interdependientes*, es decir, un número de tareas no repetitivas que es necesario realizar en un cierto orden con el fin de lograr el objetivo del proyecto.

- Un proyecto utiliza varios *recursos* para realizar las tareas. Esos recursos pueden incluir diferentes personas, organizaciones, equipos, materiales e instalaciones. Por ejemplo, una boda es un proyecto que quizá incluya recursos tales como un proveedor de banquetes, un florista, una limusina y un salón para la recepción.
- Un proyecto tiene un *marco de tiempo específico*, o tiempo limitado. Tiene un tiempo de inicio y una fecha para la cual se tiene que lograr el objetivo. Por ejemplo, la restauración de una escuela de primera enseñanza pudiera tener que terminarse entre el 20 de junio y el 20 de agosto.
- Un proyecto puede ser un *intento único*. Algunos proyectos - como diseñar y construir una estación espacial- son únicos porque nunca antes se ha intentado hacerlos. Otros proyectos, como desarrollar un nuevo producto, construir una casa, o planear una boda, son únicos debido a que se requiere que sean hechos de acuerdo a ciertas especificaciones especiales. Por ejemplo, una boda puede ser una ocasión sencilla, informal, con pocos amigos en una capilla, o un acontecimiento espectacular preparado para un príncipe.
- Un proyecto tiene un cliente. El cliente es la entidad que proporciona los fondos necesarios para el logro del proyecto; puede ser una persona, una organización, o un grupo de dos o más personas u organizaciones. Cuando un contratista construye una casa para una pareja de acuerdo a ciertas especificaciones especiales, la pareja es el cliente que proporciona los fondos para el proyecto. Cuando una compañía recibe recursos del gobierno para desarrollar un dispositivo automático para el manejo de materiales radiactivos, el cliente es la agencia del gobierno.

Cuando una compañía proporciona fondos para un equipo de sus empleados con el fin de actualizar el sistema de información o la administración de la empresa, el término *cliente* toma una definición más amplia, incluyendo no sólo a quien proporciona los medios para el proyecto (la administración de la compañía) sino también a otras personas que tienen participación en la empresa, por ejemplo las personas que serán los usuarios finales del sistema de información. La persona que administra el proyecto y su equipo tienen que cumplir con éxito el objetivo fijado para satisfacer al (los) cliente(s).

Por último, un proyecto incluye un *grado de incertidumbre*. Antes de que se inicie un proyecto se prepara un plan sobre la base de ciertos supuestos y estimados. Es importante documentar estos supuestos ya que influirán sobre el desarrollo del presupuesto, el programa y el alcance del trabajo del proyecto. Un proyecto se basa en un grupo único de tareas y estimados de qué duración debe tener cada tarea, de los recursos y supuestos sobre la disponibilidad y capacidad de esos recursos y estimados de sus costos. Esta combinación de suposiciones y estimados ocasionan un grado de incertidumbre con relación a si el objetivo del proyecto será alcanzado por completo. Por ejemplo, el alcance del proyecto quizá se logre para la fecha fijada como meta, pero el costo final puede ser mucho más alto de lo anticipado debido a los bajos estimados iniciales del costo de ciertos recursos. Según avanza el proyecto, algunas de las suposiciones serán perfeccionadas o reemplazadas con información basada en hechos. Por ejemplo, una vez que se termina el diseño conceptual del informe anual de una compañía, se puede estimar mejor la cantidad de tiempo y de esfuerzos necesarios para completar el diseño detallado y su impresión.

A continuación se presentan algunos ejemplos de proyectos:

- Poner en escena una producción teatral.
- Desarrollar e introducir un nuevo producto.
- Planear una boda.
- Diseñar y poner en práctica un sistema de computación.
- Emitir una nueva moneda de 1.00.

Modernizar una fábrica.

Consolidar dos plantas industriales.

Convertir un sótano en una sala de estar.

Ser anfitrión de una conferencia.

Diseñar y producir un folleto.

Llevar a cabo la limpieza ambiental de un lugar contaminado.

Celebrar una reunión en una escuela de segunda enseñanza.

Construir un centro comercial.

Realizar varias cirugías en una víctima de accidente.

Realizar la celebración de un centenario.

Reconstruir un pueblo después de un desastre natural.

Ser anfitrión de una cena para veinte parientes.

Diseñar un programa de internado en empresas para estudiantes de segunda enseñanza.

### 1.2.3 El medio ambiente del proyecto.

Por lo general el logro exitoso del objetivo del proyecto está limitado por cuatro factores: alcance, costo, programa y satisfacción del cliente. Véase la figura 1.2.3.1.

*El alcance de un proyecto* - conocido también como el alcance del proyecto o el alcance del trabajo-, es todo el trabajo que se tiene que realizar con el fin de que el cliente quede satisfecho de que las entregas (el producto o los artículos tangibles a proporcionarle), cumplan con los requisitos o los criterios de aceptación acordados al inicio del proyecto. Por ejemplo, el alcance del proyecto pudiera ser todo el trabajo incluido en limpiar el terreno, construir una casa y preparar los jardines de acuerdo a las especificaciones acordadas entre el contratista y el comprador. El cliente espera que el alcance del trabajo se realice con calidad. Por ejemplo, en el proyecto de construir una casa, el cliente espera que la mano de obra sea de la más alta calidad. Si se termina el alcance del trabajo, pero se dejan ventanas difíciles de abrir y cerrar, llaves que goteen, o un jardín lleno de rocas, dará como resultado un cliente descontento.

*El costo de un proyecto* es la cantidad que ha convenido pagar el cliente por las entregas aceptables del proyecto. Se basa en un presupuesto que incluye un estimado de los costos, relacionados con los diversos recursos que se usaran para realizar el proyecto. Podría incluir los sueldos de las personas que trabajarán en el proyecto, los materiales y suministros, el alquiler de equipos o instalaciones y los honorarios de los subcontratistas o asesores que realizarán algunas de las tareas del proyecto. Por ejemplo, si el proyecto es una boda, las partidas presupuestadas pueden incluir las flores, el vestido de la novia, el traje de etiqueta del novio, el proveedor del banquete, el pastel, el alquiler de la limusina, el fotógrafo, etcétera.

*El programa de un proyecto* es la relación de tiempos que especifica cuándo se debe iniciar y terminar cada actividad. Por lo general el objetivo del proyecto, expresa el tiempo en el cual se tiene que completar el alcance del proyecto en términos de una fecha específica, acordada entre el cliente y la persona o la organización que realiza el trabajo. Podría ser la fecha en que se llevará a cabo la celebración del centenario de una ciudad, o la fecha en la que se quiere terminar el añadir un salón de descanso a la casa. El objetivo de cualquier proyecto es completar el alcance dentro del presupuesto para una fecha determinada, a satisfacción del cliente. Para ayudar a asegurar el logro de este objetivo, es importante desarrollar un plan antes del inicio del proyecto; éste debe incluir todas las tareas de trabajo, los costos relacionados y los estimados del tiempo necesario para terminarlos. La carencia de este tipo de plan aumenta el riesgo de fracaso y no cumplir el alcance total del proyecto dentro del presupuesto y a tiempo. Una vez que se inicia un proyecto pueden ocurrir circunstancias imprevistas que pongan en peligro el logro del objetivo fijado con relación al alcance, el costo o el programa.

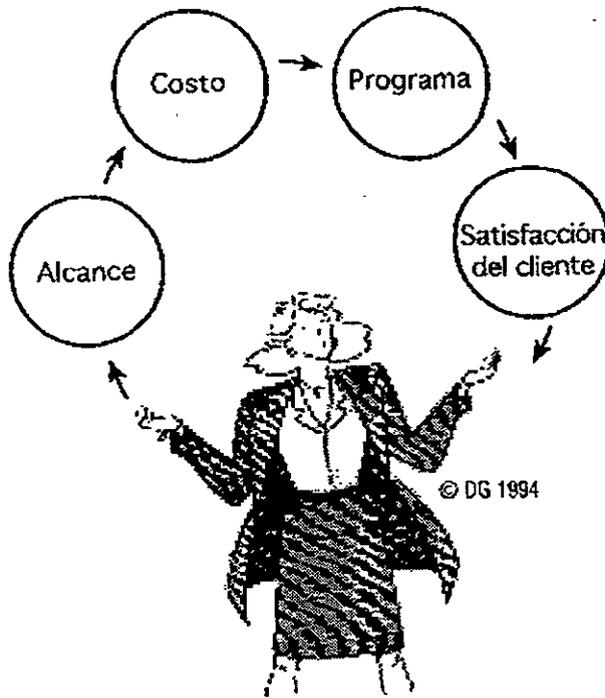


FIGURA. 1.2.3.1 Factores que restringen el éxito del proyecto

- El costo de algunos de los materiales puede ser más alto de lo estimado originalmente.
- La inclemencia del tiempo puede ocasionar una demora.
- Quizá se requiera diseñar de nuevo y hacer modificaciones a una maquinaria automática y compleja para lograr que cumpla con las especificaciones de desempeño.

El reto para el gerente de proyectos es prevenir, y superar este tipo de circunstancias, con el fin de completar el alcance del proyecto de acuerdo al programa, dentro del presupuesto y a satisfacción del cliente. La *buena planeación* y la *comunicación* son esenciales para evitar que ocurra un problema y minimizar su repercusión sobre el logro del objetivo del proyecto si sucede. El gerente del proyecto necesita ser proactivo en la planeación y en la comunicación, y debe proporcionar liderazgo a su equipo para

lograr el objetivo del mismo.

En definitiva, la responsabilidad del gerente del proyecto es asegurarse de que el cliente quede satisfecho, más que sólo completar el alcance del proyecto dentro del presupuesto y a tiempo, o preguntar al cliente si está satisfecho al final del proyecto. Requiere de una comunicación continua con el cliente para mantenerlo informado y determinar si las expectativas han cambiado. Ejemplos de formas de lograr esas comunicaciones son: reuniones programadas periódicamente o informes de avance, frecuentes discusiones por teléfono y por correo electrónico. La satisfacción del cliente significa incluirlo como un socio en el resultado exitoso del proyecto mediante su participación activa. El gerente de proyecto tiene que estar consciente del grado de satisfacción del cliente todo el tiempo. Al mantener una comunicación periódica con él, el gerente del proyecto le demuestra que está sinceramente preocupado por sus expectativas y evita desagradables sorpresas posteriores.

#### Ciclo de Vida de un Proyecto

En la figura 1.2.3.2 se muestran las cuatro fases del ciclo de Vida del proyecto y la cantidad relativa de esfuerzo y tiempo dedicados a cada fase. Según el proyecto se desplaza a través de su ciclo de vida, diferentes organizaciones, personas y recursos desempeñan papeles dominantes.

Los proyectos "nacen" cuando el cliente, las personas o la organización dispuestas a proporcionar los fondos para satisfacer la necesidad identifica una necesidad. Por ejemplo, para una familia que está creciendo en tamaño, la necesidad quizá sea una casa más amplia, mientras que para una compañía, el problema quizá sea una alta tasa de desperdicios en su proceso de fabricación, que hace que sus costos sean más altos y los tiempos de producción más largos que los de sus competidores. El cliente primero tiene que identificar la necesidad o el problema. En ocasiones el problema se identifica con rapidez, como en el caso de un desastre, como pueden ser un terremoto o una explosión. En otras situaciones, quizá se requieran meses para que el cliente

identifique con claridad una necesidad, recopile información sobre el problema y defina ciertos requisitos que tiene que cumplir la persona, el equipo del proyecto o el contratista que solucionará el problema.

Esta *primera fase* del ciclo de vida del proyecto incluye la identificación de una necesidad, un problema, o una oportunidad, y puede dar como resultado que el cliente solicite propuestas a personas, a un equipo de proyectos u organizaciones (contratistas) para resolver el problema identificado. Por lo general la necesidad y los requisitos los redacta el cliente por escrito en un documento denominado una solicitud de propuesta (SDP).

A través de la SDP el cliente les pide a personas individuales o contratistas que presenten propuestas sobre cómo solucionarían el problema, junto con el costo correspondiente y el programa. Una pareja que necesita una nueva casa quizá dedique tiempo a identificar los requisitos para la casa el tamaño, estilo, número de habitaciones, ubicación, cantidad máxima que quiere gastar y la fecha para la que le gustaría cambiarse.

Entonces se pueden poner por escrito estos requisitos y pedir a varios contratistas que proporcionen planos de casas y estimados de costos. Una compañía que ha identificado la necesidad de actualizar su sistema de computación pudiera documentar sus requisitos en una SDP y enviarla a varias empresas de asesoría de computación. Sin embargo, no todas las situaciones incluyen una SDP formal. Con frecuencia las necesidades se definen de un modo informal durante una reunión, o una discusión entre un grupo de personas. Algunas de las personas quizá entonces se ofrezcan o se les solicite que preparen una propuesta para determinar si se debe llevar a cabo un proyecto para resolver la necesidad. Se puede presentar un escenario como éste, cuando la administración de un hospital quiere establecer un centro de cuidados durante el día, en el propio hospital, para los hijos de sus empleados.

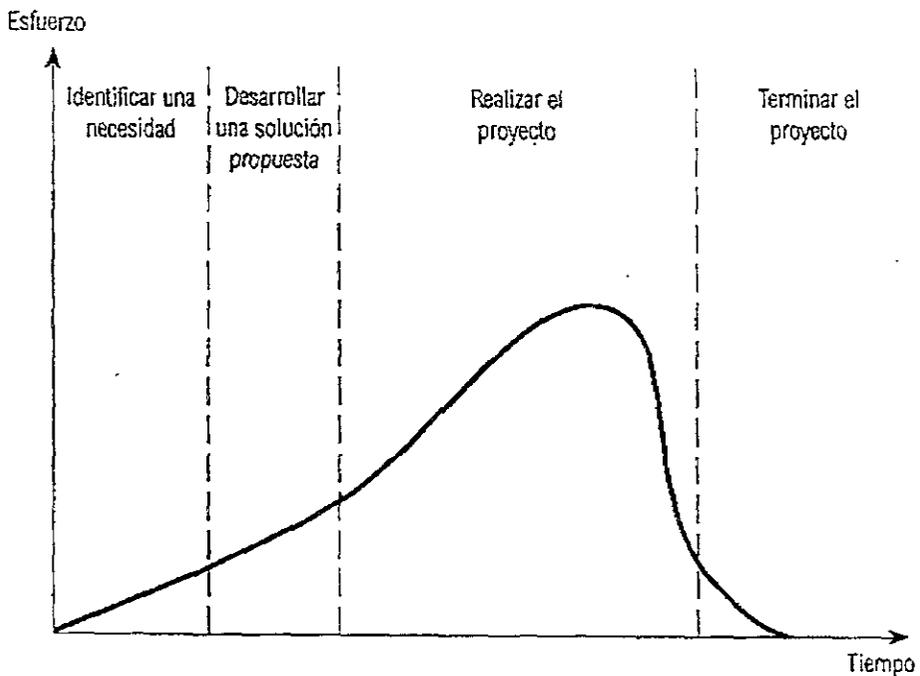


FIGURA 1.2.3.2 Ciclo de vida del proyecto

El equipo administrativo o un gerente específico puede poner por escrito los requisitos en un documento y dárselo a un equipo interno de proyectos, que a su vez entregará una propuesta de cómo establecer el centro. En este caso, el contratista es el equipo interno de proyectos del propio hospital y el cliente es el gerente del hospital o, posiblemente, el consejo de dirección.

Es importante definir la necesidad correcta. Por ejemplo, ¿es la necesidad proporcionar un centro de cuidados durante el día en el propio hospital, o es proporcionar atención infantil a los hijos de los empleados del hospital? ¿Es por necesidad "en el propio hospital" necesariamente parte de la necesidad?

La *segunda fase* del ciclo de vida del proyecto es el desarrollo de una solución propuesta a la necesidad o al problema. Esta fase da como resultado la presentación

de una propuesta al cliente por parte de una o más personas u organizaciones (contratistas), quienes desearían que el cliente les pagara por poner en práctica posteriormente la solución propuesta. En esta fase el esfuerzo del contratista es predominante. Los contratistas interesados en contestar a la SDP quizá dediquen varias semanas a desarrollar enfoques para solucionar el problema, estimar los tipos y cantidades de recursos que sean necesarios, y estimar el tiempo que haría falta para diseñar y poner en práctica la solución propuesta. Cada contratista documenta esta información en una propuesta por escrito y la entrega al cliente. Por ejemplo, varios contratistas pueden presentar propuestas a un cliente para desarrollar y poner en práctica un sistema automatizado de facturación y cobranza. Después de que el cliente evalúa las presentaciones y selecciona la propuesta ganadora, el cliente y el contratista ganador negocian y firman un **contrato** (acuerdo). En muchas situaciones la solicitud de propuesta quizá no incluya solicitar propuestas competitivas de contratistas externos. El equipo interno de proyectos de la propia compañía quizá desarrolle una propuesta en respuesta a una necesidad o solicitud definida por la administración. En este caso, el proyecto lo llevarían a cabo los propios empleados de la compañía en lugar de un contratista externo.

La *tercera fase* del ciclo de vida del proyecto es la puesta en práctica de la solución propuesta. Esta fase se inicia después de que el cliente decida cuál de las soluciones propuestas satisface mejor la necesidad y se llegue a un acuerdo entre el cliente y la persona o el contratista que presentó la propuesta. Esta fase, conocida como desarrollar el proyecto, incluye hacer la planeación detallada del proyecto y después poner en práctica ese plan para lograr el objetivo del proyecto. Durante esta etapa se utilizarán diferentes tipos de recursos. Por ejemplo, si el Proyecto es diseñar y construir un edificio para oficinas, el esfuerzo del proyecto quizá primero incluya a varios arquitectos e ingenieros para desarrollar los planos del edificio. Entonces, según se inicia la construcción, los recursos necesarios aumentarán en forma importante para incluir trabajadores del acero, carpinteros, electricistas, pintores, etc. El proyecto se dará por terminado después de que se complete el edificio y un número más pequeño de diferentes trabajadores terminarán los jardines y darán los toques finales al interior.

Esta fase da como resultado el logro del objetivo del proyecto, dejando al cliente satisfecho de que el alcance total del trabajo se completó con calidad, dentro del presupuesto y a tiempo. Por ejemplo, la tercera parte queda completa cuando un contratista ha terminado el diseño y la instalación de un sistema de automatización, de acuerdo a las necesidades específicas del cliente, que pase en forma satisfactoria pruebas de desempeño y sea aceptado por el cliente o cuando un equipo interno de proyectos de una compañía ha terminado un proyecto, como respuesta a una solicitud de la administración, que consolidó dos de sus instalaciones en una. *La fase final* del ciclo de vida del proyecto es terminario. Cuando un proyecto está terminado se necesita realizar ciertas actividades de cierre, por ejemplo, confirmar que todas las entregas se han hecho al cliente y han sido aceptadas por él, que se han cobrado todos los pagos y que se han pagado todas las facturas. Durante esta fase, una tarea importante es evaluar el desempeño del proyecto con el fin de aprender qué se pudiera mejorar si se llevara a cabo un proyecto similar en el futuro. Esta fase debe incluir, obtener retroalimentación del cliente para determinar su nivel de satisfacción y si el proyecto cumplió con sus expectativas. También se debe obtener retroalimentación del equipo del proyecto en la forma de recomendaciones, para mejorar el desempeño de proyectos en el futuro.

Los ciclos de vida de los proyectos varían en duración desde algunas semanas hasta varios años, dependiendo del contenido, complejidad y magnitud del proyecto. Es más, no todos los proyectos pasan formalmente a través de las cuatro fases del ciclo de vida del proyecto. Si un grupo de voluntarios de la comunidad decide que quiere utilizar su propio tiempo, habilidades y recursos para organizar una campaña para obtener alimentos para las personas sin hogar, puede comenzar directamente en la fase tres, planeación de la actividad, y llevarla a cabo. Las primeras dos fases del ciclo de vida no serían importantes para este tipo de proyecto. En igual forma, si el gerente general de una compañía determina que el cambiar la disposición de los equipos en la fábrica aumentará la eficiencia, simplemente pudiera dar instrucciones al gerente de producción, para que inicie este proyecto y lo ponga en práctica, utilizando el personal propio de la compañía. En este caso no habría una solicitud por escrito de una

propuesta de contratistas externos.

En otras situaciones, por ejemplo, el proyecto de remodelar una casa en que probablemente se utilizará un contratista, el cliente quizá pase por las primeras dos fases del ciclo de vida del proyecto en una forma menos estructurada, más informal. Quizá no ponga por escrito todos los requisitos ni solicite estimados a varios contratistas. En lugar de ello, pudiera llamar a un contratista que le haya hecho antes un trabajo satisfactorio, a él o a algún vecino, le explique lo que quiere que se haga y le pida que le proporcione algunos bosquejos y un estimado de costos. En general, el ciclo de vida del proyecto se sigue de manera más formal y estructurada cuando el proyecto se realiza en un ambiente de negocios. Tiende a ser menos formal cuando el proyecto lo lleva a cabo una sola persona o un grupo de voluntarios.

#### Proceso de la Administración del Proyecto

Dicho en forma breve, el proceso de administración del proyecto significa *planear el trabajo y después trabajar el plan*. Un grupo de entrenadores quizá dedique horas a preparar planes exclusivamente para un juego; después el equipo los ejecuta para intentar cumplir con el objetivo, la victoria. En forma similar, la administración de proyectos incluye primero *establecer un plan y después llevar a cabo ese plan*, para lograr el objetivo del proyecto.

El esfuerzo principal en la administración de un proyecto tiene que estar centrado en establecer un plan de línea base, que proporcione un plan de ruta para indicar cómo se logrará el alcance del proyecto a tiempo y dentro del presupuesto. Este esfuerzo de planeación incluye los pasos siguientes:

1. **Definir con claridad el objetivo del proyecto.** La definición tiene que ser aceptada por el cliente y la persona o la organización que realizará el proyecto.
2. **Dividir y subdividir el alcance del proyecto en "piezas" importantes, o pa-**

**quetes de trabajo.** Aunque los proyectos trascendentes quizá parezcan abrumadores cuando se contemplan como un conjunto, una forma de resolverlos es dividirlos en partes. Una *estructura de división del trabajo* (EDT) es un árbol jerárquico de elementos o partidas de trabajo, logradas o producidas por el equipo del proyecto durante el proyecto. Por lo general la estructura de división del trabajo identifica a la organización o a la persona que tiene la responsabilidad de cada paquete de trabajo.

3. **Definir las actividades específicas** que son necesarias de realizar para cada paquete de trabajo con el fin de lograr el objetivo del proyecto.
4. **Presentar gráficamente las actividades bajo la forma de un diagrama de red.** Este diagrama muestra el orden necesario y las interdependencias de las actividades para lograr el objetivo del proyecto. La figura 1.2.3.4 es un ejemplo de un diagrama de red.
5. **Hacer un estimado de tiempo de la duración que tendrá que completar cada actividad.** También es necesario determinar qué tipos de recursos y cuánto de cada recurso se necesita para terminar cada actividad dentro de la duración estimada.
6. **Hacer un estimado de costos para cada actividad.** El costo se basa en los tipos y cantidades de recursos necesarios para cada actividad.
7. **Calcular el programa y el presupuesto de un proyecto,** para determinar si el mismo se puede terminar dentro del tiempo requerido, con los fondos asignados y con los recursos disponibles. Si no es así, se tienen que hacer ajustes al alcance del proyecto, a los tiempos estimados de las actividades, o a las asignaciones de recursos hasta que se pueda establecer un **plan** de línea base alcanzable, realista (un mapa de ruta para lograr el alcance del proyecto a tiempo y dentro del presupuesto). En la figura 1.2.3.5 se muestra un ejemplo de un programa de

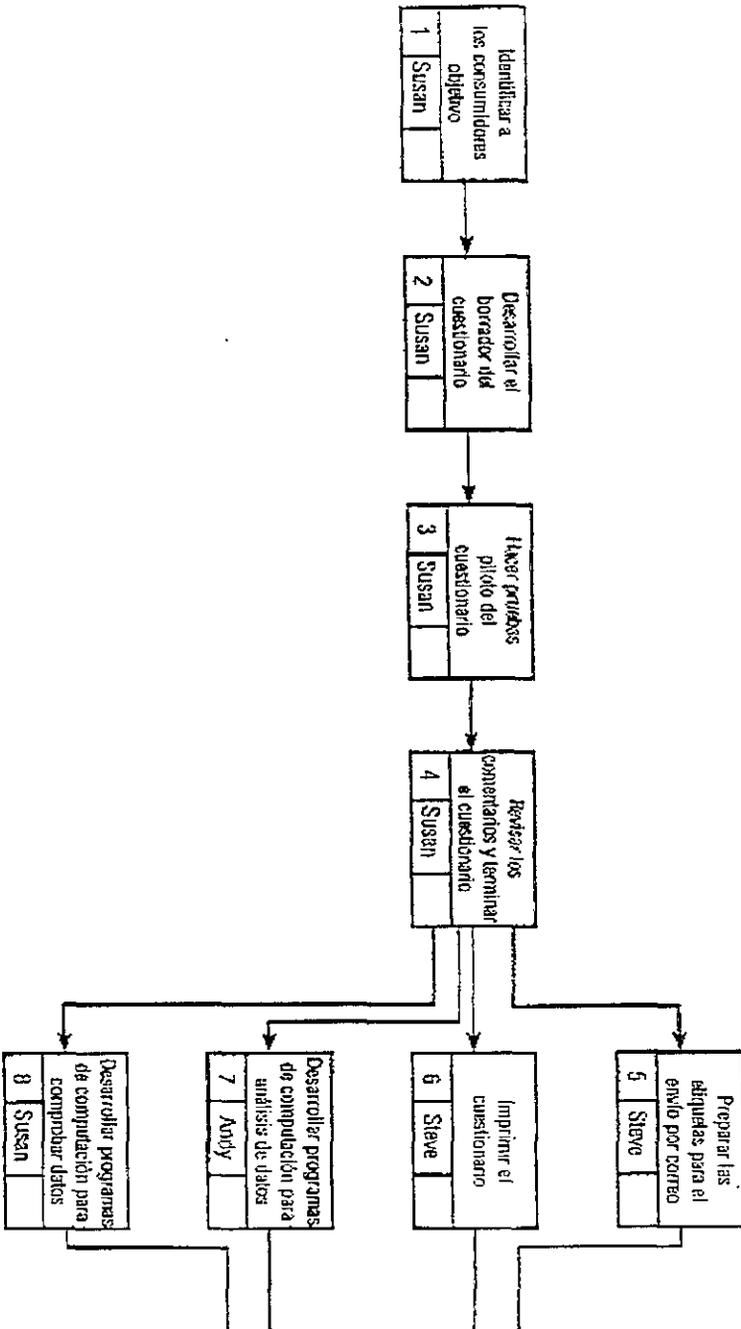
proyecto y en la figura 1.2.3.6 se muestra un presupuesto de proyecto.

La planeación determina qué se necesita hacer, quién lo hará, cuánto tiempo se necesitará y cuánto costará. El resultado de este esfuerzo es un plan de línea base. El tomar el tiempo necesario para desarrollar un plan bien pensado es crítico para el logro exitoso de cualquier proyecto. Muchos proyectos han excedido sus presupuestos, incumplido sus fechas de terminación o han cumplido sus requisitos sólo en forma parcial, debido a no contar con un plan de línea base viable antes de que se iniciara el proyecto.

El plan de línea base para un proyecto se puede mostrar en un formato gráfico o tabular para cada periodo (semana, mes), desde el inicio del proyecto hasta su terminación. La información debe incluir:

- Las fechas de inicio y terminación de cada actividad
- Las cantidades de los diversos recursos que se necesitarán durante cada periodo.
- El presupuesto para cada periodo, así como el presupuesto acumulado desde el inicio del proyecto a través de cada periodo

Una vez que se ha establecido un plan de línea base, se tiene que poner en práctica. Esto incluye realizar el trabajo de acuerdo al plan y controlar el trabajo en forma tal, que el alcance del presupuesto se logre dentro del presupuesto y el programa, a satisfacción del cliente.



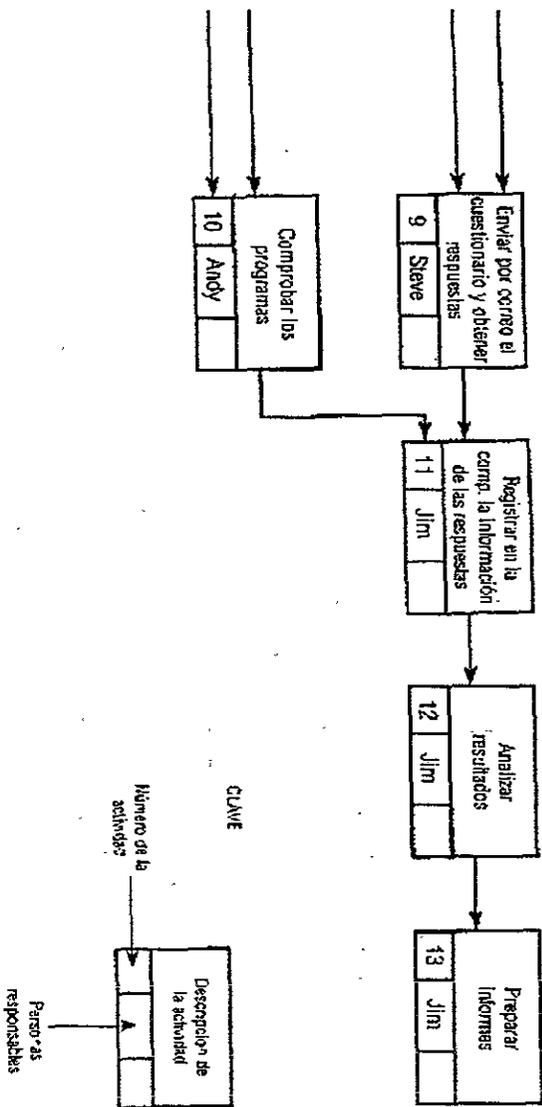


FIGURA 1.2.3.4 Diagrama de Red

Una vez que se inicia el proyecto es necesario supervisar el avance, para asegurar que todo vaya de acuerdo al plan. En esta etapa, el proceso de administración del proyecto incluye medir la evolución real y compararlo con el planeado. Para medir el progreso real es importante estar informado de cuáles actividades se han iniciado realmente y/o terminado, cuándo se iniciaron y/o terminaron y cuánto dinero se ha gastado o comprometido. Si durante algún momento del proyecto, la comparación del progreso real con el planeado revela que el proyecto está atrasado, que ha excedido el presupuesto, o que no cumple con las especificaciones técnicas, se tiene que llevar a cabo una acción correctiva, para hacer que el proyecto vuelva a estar dentro de las especificaciones.

	ACTIVIDAD	RESPON.	DUR. ESTIM.	FECHA MAS TEMPRANA		FECHA MAS TARDIA		HOLGURA TOTAL
				DE INICIO	DE TERMIN.	DE INICIO	DE TERMIN.	
1	Identificar a los consumidores objetivo	Susan	3	0	3	-3	-5	-8
2	Desarrollar el borrador de un cuestionario	Susan	10	3	13	-5	5	-8
3	Hacer pruebas piloto del cuestionario	Susan	20	13	33	5	25	-8
4	Revisar los comentarios y terminar el cuestionario	Susan	5	33	38	25	30	-8
5	Preparar las etiquetas para el envío por correo	Steve	2	38	40	38	40	0
6	Imprimir el cuestionario	Steve	10	38	48	30	40	-8
7	Desarrollar programas de computación para análisis de datos	Andy	12	38	50	88	100	50
8	Desarrollar programas de computación para comprobar los datos	Susan	2	38	40	98	100	60
9	Enviar por correo el cuestionario y obtener respuestas	Steve	65	48	113	40	105	-8
10	Comprobar los programas de computación	Andy	5	50	55	100	105	50
11	Registrar información de las respuestas	Jim	7	113	120	105	112	-8
12	Analizar los resultados	Jim	8	120	128	112	120	-8
13	Preparar informes	Jim	10	128	138	120	130	-8

FIGURA 1.2.3.5 Programa del proyecto

Antes de que se tome una decisión para poner en práctica una acción correctiva, quizá sea necesario evaluar varias acciones alternativas, para asegurarse de que la acción correctiva hará que de nuevo el proyecto esté dentro del alcance, el tiempo y las limitaciones del presupuesto del objetivo. Por ejemplo, hay que estar conscientes de que añadir recursos para compensar el tiempo y volver a estar dentro del programa, quizá dé como resultado excederse del presupuesto planeado. Si un proyecto queda demasiado fuera de control, quizá sea difícil lograr el objetivo del proyecto sin sacrificar el alcance, el presupuesto, el programa o la calidad.

La clave para el control efectivo del proyecto es medir el progreso real y compararlo con el planeado sobre una base oportuna y periódica y, si es necesario, realizar la acción correctiva de inmediato. El confiar que un problema desaparecerá sin la intervención correctiva es de ingenuos. Con base en el avance real es posible pronosticar un programa y un presupuesto para la terminación del proyecto. Si estos parámetros se encuentran más allá de los límites del objetivo del proyecto, de inmediato se deben poner en práctica las acciones correctivas necesarias.

El intentar realizar un proyecto sin establecer primero un plan de línea base es arriesgado. Es como comenzar unas vacaciones sin un mapa de carreteras, itinerario ni presupuesto. Se puede terminar en un lugar remoto, ¡sin dinero y sin tiempo!.

### Beneficios de la Administración de Proyectos

El beneficio definitivo de poner en práctica técnicas de administración de proyectos es tener un *cliente satisfecho*, tanto si usted es el cliente de su propio proyecto, por ejemplo remodelar su sótano, como si lo es una empresa (contratista), a quien un cliente le paga para realizar un proyecto. El completar el alcance total del proyecto con calidad, a tiempo y dentro del presupuesto, proporciona una gran sensación de satisfacción. Para un contratista significa que puede llevarlo en el futuro a negocios adicionales con el mismo cliente o a nuevos negocios recomendados por clientes previamente satisfechos.

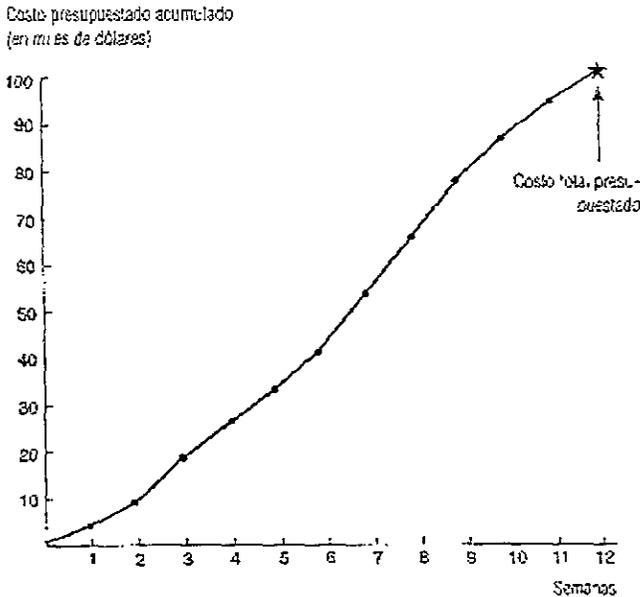


FIGURA 1.2.3.6 Curva del costo presupuestado acumulado

"¡Oiga! Eso está muy bien para el cliente, pero ¿y qué me dice de mí? ¿Qué hay para mí?" Si usted es el gerente del proyecto, tiene la satisfacción de haber dirigido un esfuerzo de proyecto exitoso. También ha resaltado su reputación como gerente de proyectos y se ha puesto en una posición de obtener mayores oportunidades de carrera. Si usted es miembro de un equipo de proyectos que completó con éxito un proyecto, siente la satisfacción de encontrarse en un equipo ganador. No sólo contribuyó al éxito del proceso sino que también probablemente amplió sus conocimientos y mejoró sus habilidades en el proceso. Si eligió permanecer como un participante individual, estará en posibilidad de hacer una mayor contribución a proyectos futuros, más complicados. Si está interesado, con el tiempo, al administrar proyectos, se encontrará en una posición de tomar responsabilidades de proyectos adicionales.

¡Cuándo los proyectos son exitosos, todos ganan!

## 1.2.4 Tipos de proyectos

Puede decirse que no existen diferentes tipos de proyectos, y estos unicamente pueden clasificarse de acuerdo a:

- a) El tipo de proyecto final.
- b) El número de empleados que participan.
- c) El capital invertido.

## 1.2.5 Clasificación de proyectos

a) Clasificación de acuerdo al tipo de proyecto final:

1. Proyectos con un fin de consumo(producto): casas, maqunaria, automoviles, aparatos eléctricos, etc.
2. Proyectos con un fin de servicio: consultoría, educación, servicios financieros, diversiones, etc.

b) Clasificación de acuerdo al número de empleados.que participan:

1. Pequeña (hasta 20 personas)
2. Mediano (21 a 500 personas)
3. Grande (más de 500 personas)

c) De acuerdo al capital invertido:

El capital invertido puede variar de acuerdo al tipo de proyecto, al número de personas involucradas, a la complejidad del proyecto o a la magnitud.

## 1.2.6 Fase organizacional

### 1.2.6.1 Planeación

La tercera fase del ciclo de vida del proyecto tiene dos partes: hacer su planeación

detallada y después poner en práctica ese plan, para lograr el objetivo del mismo. Antes de precipitarse e iniciar el proyecto en sí, el contratista o el equipo tienen que dedicar tiempo suficiente a planear en forma apropiada el proyecto. Es necesario preparar un programa o un plan general que muestre cómo se realizarán las tareas dentro del presupuesto y en el tiempo señalado. El intentar realizar un proyecto sin un plan es como intentar armar la bicicleta de un niño sin leer primero las instrucciones. Las personas que piensan que la planeación es innecesaria o que es una pérdida de tiempo, invariablemente después, necesitarán dedicar más tiempo para volver a hacer las cosas. Es importante *planear el trabajo y después trabajar el plan*. De lo contrario, el resultado será caos y frustración y el riesgo de fracaso será más alto.

La parte de la planeación incluye elaborar con gran detalle el plan, el programa y el presupuesto. Por lo general durante la fase de la propuesta (segunda) no se justifican el tiempo ni el gasto que se requieren para hacer una planeación detallada.

1. *Definir con claridad el objetivo del proyecto.* La definición tiene que ser aceptada por el cliente y la persona u organización que realizará el proyecto.
2. *Dividir el alcance del proyecto en "piezas" importantes, o paquetes de trabajo.* Aunque los proyectos importantes quizá parezcan ser abrumadores cuando se contemplan como un conjunto, una forma de dominar la empresa más monumental es dividirla. La estructura de división del trabajo es un árbol jerárquico de los elementos de trabajo o partidas que realiza o produce el equipo durante el proyecto. Con frecuencia la estructura de división del trabajo identifica la organización o la persona individual responsable de cada paquete de trabajo.
3. *Definir las actividades específicas que son necesarias de realizar en cada paquete de trabajo con el fin de lograr el objetivo del proyecto.*
4. *Presentar en forma gráfica las actividades bajo la forma de un diagrama de red.* Este diagrama muestra el orden necesario y las interdependencias de las

actividades para alcanzar el objetivo del proyecto.

5. *Calcular el tiempo estimado que requerirá completar cada actividad. También es necesario determinar los tipos de recursos y la cantidad que se requiere de cada uno de ellos para terminar cada actividad dentro del tiempo estimado.*
6. *Calcular el costo estimado para cada actividad. El costo se basa en los tipos y cantidades de recursos que se requieren para cada actividad.*
7. *Calcular un programa y un presupuesto para el proyecto para determinar si se puede lograr dentro del tiempo requerido, con los fondos asignados y con los recursos disponibles. Si no es así, se tienen que hacer ajustes al alcance del proyecto, a los tiempos estimados de las actividades o las asignaciones de recursos, hasta que se pueda establecer un plan de línea base (un programa para lograr el alcance del proyecto a tiempo y dentro del presupuesto), que sea factible y realista.*

La planeación determina lo que se necesita hacer, quién lo hará, cuánto tiempo se necesitará y cuánto costará. El resultado de este esfuerzo es un plan de línea base. El dedicar tiempo a desarrollar un plan bien pensado es crítico para el logro exitoso de cualquier proyecto. Porque muchos de ellos han excedido sus presupuestos, fallado en las fechas de terminación, o han satisfecho sólo en forma parcial sus especificaciones técnicas, debido a que no existía un plan de línea base viable antes de iniciar el proyecto.

Es importante que las personas que participarán en la realización del proyecto también colaboren en la planeación del trabajo. Por lo general son los que conocen más sobre qué actividades detalladas son necesarias de realizar. Incluso, al contribuir en la planeación del trabajo, estas personas se comprometen a lograrlo de acuerdo al plan.

La participación crea compromiso.

## 1.2.7 Fase de construcción

### 1.2.7.1 Realización

Una vez que se ha desarrollado el plan de línea base se puede proceder al trabajo del proyecto. El equipo, dirigido por el gerente del proyecto, pondrá en práctica el plan y realizará las actividades o elementos de trabajo de acuerdo con él. El ritmo de la actividad aumentará según se vayan incluyendo más y diversos recursos al desarrollo de las tareas del proyecto.

Realizar el proyecto, poner en práctica la solución propuesta, es la tercera fase de su ciclo de vida, como se muestra en la figura 1.2.7.1

Para el proyecto de realizar un festival en la ciudad, entre los principales elementos de trabajo se pueden incluir los siguientes:

1. Preparar la publicidad - anuncios en los periódicos, carteles, boletos, etcétera.
2. Seleccionar voluntarios.
3. Organizar juegos, incluyendo la construcción de puestos y adquisición de premios.
4. Contratar los juegos mecánicos y obtener los permisos necesarios.
5. Identificar artistas que *entretengan y construir el escenario y la tribuna para el público.*
6. Hacer los arreglos para la comida, desde su preparación hasta la construcción de *los puestos para concesiones.*
7. Organizar todos los servicios de respaldo, como el estacionamiento, la limpieza, la seguridad y las instalaciones de baños.

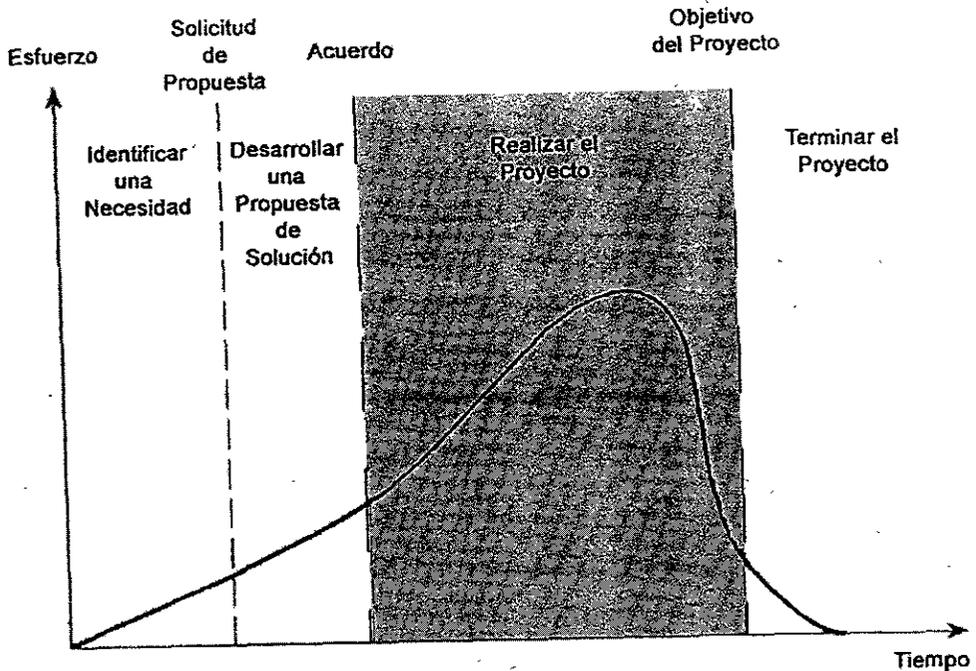


FIGURA 1.2.7.1 Ciclo de vida del proyecto

Para el proyecto más técnico de diseñar, construir, e instalar una máquina para envasado, especializada, automatizada, de alta velocidad, en la fábrica del cliente, los elementos de trabajo importantes pueden incluir lo siguiente:

1. Desarrollar los diseños, tanto preliminares como detallados, incluyendo la preparación de especificaciones, dibujos, gráficas de flujo y una relación de materiales.
2. Preparar planes para que el contratista pruebe los componentes, subsistemas y sistemas, tanto antes de embarcar el equipo a la planta del cliente como después de haber sido instalado, para asegurar que el equipo cumple con los requisitos del cliente, quien quizá quiera revisar y aprobar los planes de pruebas antes de que se inicien.
3. Realizar reuniones de revisión de diseños, tanto internamente como con el cliente.

Con base en estas reuniones el cliente puede iniciar o aprobar cambios a la propuesta original. Estos cambios podrían tener una repercusión sobre el alcance, el programa y el precio. Puede ser necesario que el cliente modifique el contrato, y el contratista quizá tenga que planear de nuevo el proyecto para incluir cualquier cambio.

4. Pedir los materiales y las piezas.
5. Fabricar los componentes y las piezas.
6. Diseñar y probar los programas de computación.
7. Montar y probar los equipos, incluyendo la prueba de los componentes, montarlos en submontajes, probar cada uno y probar todo el sistema de equipos.
8. Integrarlos equipos y los programas de computación y probar el sistema. Los representantes del cliente quizá quieran presenciar y documentar los resultados de la prueba, para estar seguros de que cumplen con las especificaciones del contrato.
9. Preparar los requisitos de instalación, como los planos de piso y las condiciones de servicios (eléctricos, de plomería, etc.) e identificar las partidas que serán responsabilidad del cliente durante la instalación.
10. Preparar materiales de capacitación (manuales, videocintas, simulaciones en computadoras), para habilitar al cliente en la operación y mantenimiento del nuevo equipo.
11. Embarcar el equipo a la fábrica del cliente e instalarlo.
12. Realizar la capacitación del cliente.
13. Realizar las pruebas finales de aceptación, para mostrar que el equipo cumple con todos los requisitos especificados por el cliente.

#### 1.2.7.2 Control

Mientras se está realizando el proyecto, es necesario supervisar el avance para asegurar que todo vaya de acuerdo al plan. Esto incluye medir el progreso real y compararlo con el planeado. Para medir el avance real es importante mantener un seguimiento de cuáles actividades ya se han iniciado y/o terminado, cuándo lo hicieron y cuánto dinero se ha gastado o comprometido. Si en cualquier momento del proyecto

la comparación del avance real con el programado muestra que se está retrasado de acuerdo al plan, que está excediendo al presupuesto, o que no cumple con las especificaciones técnicas, se tienen que llevar a cabo acciones correctivas para hacer que el proyecto esté de nuevo de acuerdo a lo planeado.

Antes de tomar la decisión de poner en práctica una acción correctiva, quizá sea necesario evaluar varias alternativas para asegurar que ésta hará que el proyecto quede de nuevo dentro del alcance, el tiempo y las restricciones presupuestales del objetivo. Por ejemplo, se debe estar consciente de que aumentar recursos para compensar el tiempo y volver a estar dentro de lo programado, quizá dé como resultado exceder el presupuesto planeado. Si un proyecto queda demasiado fuera de control, tal vez sea difícil obtener su objetivo sin sacrificar el alcance, el presupuesto, el programa o la calidad.

La clave para el control del proyecto efectivo es medir el progreso real y compararlo con el planeado sobre una base periódica y oportuna y llevar a cabo la acción correctiva de inmediato si es necesaria. El confiar que un problema desaparecerá sin una intervención correctiva es ingenuo. Mientras más temprano se identifique y corrija la dificultad es mejor. Con base en el avance real, es posible pronosticar un programa y un presupuesto para la terminación del proyecto. Si estos parámetros están fuera del objetivo, es necesario poner en práctica de inmediato acciones correctivas.

El proceso de control del proyecto incluye recopilar información periódicamente sobre el desempeño del proyecto, y comparar el avance real con el planeado para llevar a cabo acciones correctivas si el desempeño real es inferior al planeado. Este proceso tiene que ocurrir con periodicidad.

En la figura 1.2.8.1 se indican los pasos en el proceso de control del proyecto. Se inicia con el establecimiento de un plan de línea base que muestra cómo se logrará el alcance del proyecto (tareas) a tiempo (programas), y dentro del presupuesto (recursos, costos). Una vez que el cliente y el contratista o el equipo están de acuerdo

con este plan de línea base, se puede iniciar el proyecto.

Se debe establecer un periodo de presentación de informes sistemático, para comparar el avance real con el planeado. La presentación de los informes puede ser diaria, semanal, quincenal o mensual, dependiendo de la complejidad o de la duración global del proyecto. Si se espera que tendrá una duración total de un mes, el periodo de presentación de informes puede ser tan corto como cada día.

Por otra parte, si se espera que el proyecto tenga una duración de cinco años, el periodo de presentación de informes pudiera ser de un mes.

Durante cada periodo de presentación de informes es necesario recopilar dos clases de datos o información:

1. *Datos sobre el desempeño real.* Esto incluye:

- El tiempo real en que se iniciaron y/o terminaron las actividades.
- Los costos reales gastados y comprometidos.

2. *Información sobre cualquier cambio en el alcance, el programa y el presupuesto del proyecto.*

Estos cambios los puede iniciar el cliente o el equipo del proyecto, o pueden ser el resultado de un acontecimiento inesperado, como un desastre natural, una huelga laboral o la renuncia de un miembro clave del equipo.

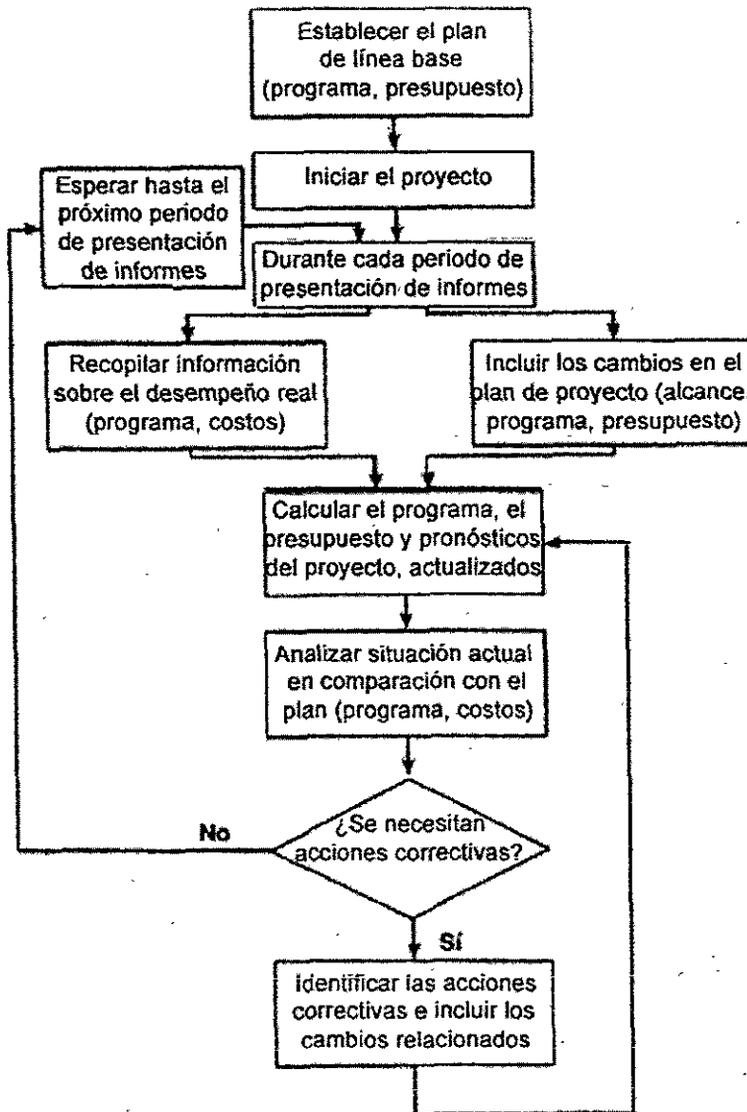


FIGURA 1.2.8.1 Proceso del Control del Proyecto

Se debe observar que, una vez que se incluyen los cambios y son aceptados por el cliente se tiene que elaborar un nuevo plan de línea base. El alcance, el programa y el presupuesto quizás sean diferentes a los del plan de línea base original.

Es crucial que los datos y la información a que se hizo referencia antes sean recopilados en forma oportuna y sean utilizados para calcular un programa y un presupuesto actualizados del proyecto. Por ejemplo, si los informes se preparan mensualmente, los datos se deben obtener lo más tarde posible en ese periodo mensual, con el fin de que, cuando se calculen un programa y un presupuesto actualizados, se basen en la información más tardía posible; en otras palabras, el gerente del proyecto no debe recopilar datos al inicio del mes y después esperar al fin del mes para calcular un programa y un presupuesto actualizados, porque los datos serán obsoletos y quizás ocasionen que se tomen decisiones incorrectas sobre la situación del proyecto y las acciones correctivas.

Una vez que se han calculado un programa y una cotización real, es necesario compararlos con el programa de línea base y el presupuesto para analizarlos en busca de variaciones y así determinar si el proyecto va adelantado o retrasado, y si ha excedido el costo planeado o se encuentra por debajo del mismo. Si la situación del proyecto es correcta, no son necesarias acciones correctivas; la situación se analizará de nuevo para el siguiente periodo de presentación de informes.

Sin embargo, si se determina que son necesarias acciones correctivas, se tienen que tomar decisiones con relación a cómo revisar el programa o el presupuesto.

Con frecuencia estas resoluciones incluyen un compromiso de tiempo, costo y alcance. Por ejemplo, el reducir la duración de una actividad quizá requiera aumentar costos para pagar más recursos o reducir el alcance de la tarea (y posiblemente no cumplir con los requisitos técnicos del cliente). En forma similar, al reducir los costos del proyecto, quizá requiera usar materiales de calidad inferior a la que se planeó originalmente. Una vez que se toma una decisión sobre cuáles acciones correctivas se deben llevar a cabo, se tienen que incluir en el programa y en el presupuesto. Es necesario determinar si las medidas correctivas planeadas dan como resultado un programa y un presupuesto aceptables. Si no es así, se necesitarán revisiones adicionales.

El proceso de control del presupuesto continúa a través de la tercera fase del ciclo de vida del proyecto. En general, mientras más corto sea el periodo de presentación de informes, son mejores las posibilidades de detectar problemas por anticipado y llevar a cabo acciones correctivas adecuadas. Como se mencionó antes, si un proyecto queda demasiado fuera de control, quizá sea difícil alcanzar sus objetivos sin sacrificar el alcance, el presupuesto, el programa, o la calidad.

Pueden presentarse situaciones en que sea sensato aumentar la frecuencia de las presentaciones de informes hasta que el proyecto esté de nuevo dentro de lo planeado. Por ejemplo, si un proyecto a cinco años, con presentación de informes mensuales, está en peligro por desviaciones en el programa o un creciente exceso del presupuesto, quizá sea prudente reducir el periodo de presentación de informes a una semana con el fin de supervisar más estrechamente el proyecto y las repercusiones de las acciones correctivas.

El proceso de control del proyecto es una parte importante y necesaria de su desarrollo. El solo hecho de establecer un plan de línea base sólido no es suficiente, puesto que incluso los planes mejor preparados no siempre funcionan. *La administración de proyectos es un enfoque proactivo al control de un proyecto*, para asegurar que se logre su objetivo incluso cuando las cosas no salen de acuerdo al plan.

Esta tercera fase del ciclo de vida termina cuando el cliente queda satisfecho de que se han cumplido todos los requisitos y que se ha logrado el objetivo del proyecto.

### **1.2.7.3 Terminación**

La cuarta y última fase del ciclo de vida del proyecto es terminarlo. Se inicia después de que se ha completado el trabajo del proyecto, tal como se muestra en la figura 1.2.9 e incluye varias acciones para cerrar idóneamente el proyecto.

El propósito de terminar apropiadamente un proyecto es aprender de la experiencia ganada en el mismo, con el fin de mejorar el desempeño en el futuro. Por lo tanto, las actividades relacionadas con la terminación del proyecto se deben identificar e incluir en el plan de línea base - no se deben manejar simplemente como ideas espontáneas posteriores -. Estas actividades pudieran incluir organizar y archivar los documentos del proyecto, recibir y hacer los pagos finales y realizar reuniones de evaluación posteriores a la terminación del proyecto, tanto dentro de la organización del contratista como en la del cliente.

La fase de terminación se inicia cuando se completa la realización del proyecto y el cliente acepta el resultado. En algunas situaciones esto pudiera ser un acontecimiento más bien formal, en el que un sistema automatizado cumple con un grupo de criterios o pasa pruebas que se especificaron en el contrato.

Otros proyectos, como, un fin de semana de actividades en la universidad para regresar al hogar, se completan simplemente con el transcurso del tiempo.

Cuando un contratista termina un proyecto para un cliente, tiene que verificar que se proporcionaron, de hecho, todas las partidas acordadas a entregar. Estas partidas pueden incluir capacitación o manuales de procedimientos, dibujos, gráficas de flujos, equipos, programas de computación, folletos, informes y datos. Durante la terminación del proyecto el contratista, o la organización que lo realizó, debe asegurarse de que las copias de la documentación apropiada y relacionada con el proyecto estén adecuadamente organizadas y archivadas para que se puedan recuperar con facilidad para su uso más adelante si son necesarias. En el futuro el contratista quizá quiera utilizar alguna información real de costos y programas de este proyecto terminado para que le ayuden a desarrollar el programa y los costos estimados para un proyecto propuesto.

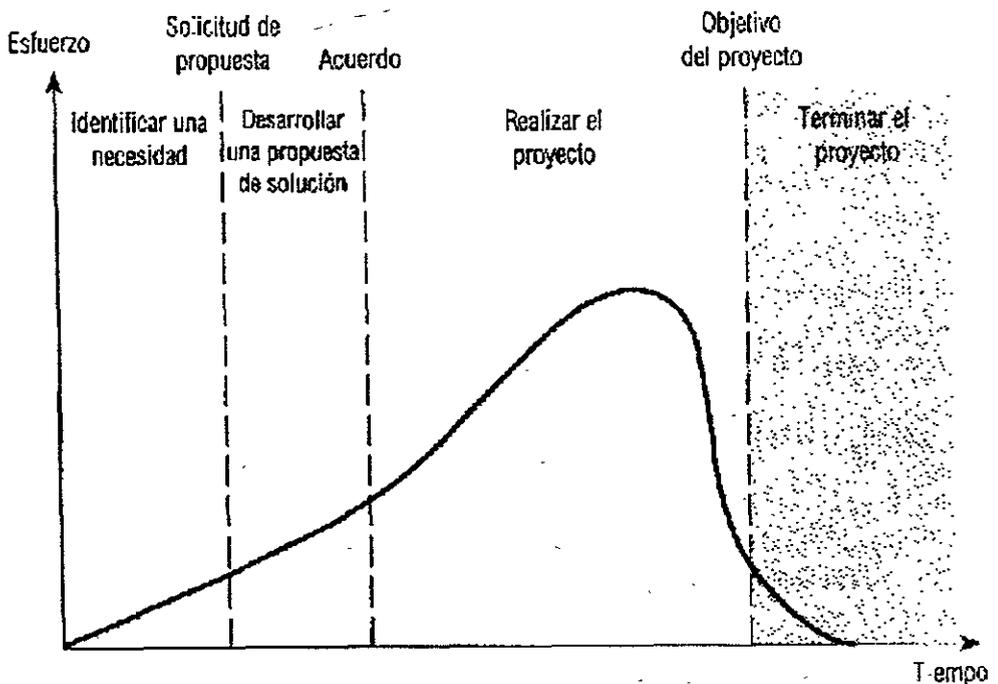


FIGURA 1.2.9.1 Ciclo de vida del proyecto

O si de lo que se trata es, por ejemplo, de organizar un festival artístico, el equipo del proyecto debe organizar toda su documentación - incluyendo sugerencias para mejorar aspectos del festival- para ser usada por el equipo que hará el festival al año siguiente. Otra actividad que se tiene que realizar durante la fase de terminación es asegurarse de que se hayan cobrado todos los pagos del cliente. Muchos contratos incluyen una cláusula de pagos por avances, donde se expresa que el cliente hará el pago final a la terminación del proyecto. En algunos casos, el pago final es un porcentaje alto del precio total - por ejemplo, el 25% -. En forma similar, se debe verificar que se hayan hecho todos los pagos a los subcontratistas o asesores y los correspondientes a la compra de cualquier clase de materiales o artículos. Una vez que se han recibido y efectuado todos los pagos, se pueden cerrar los libros o registros contables y se puede preparar un análisis financiero del proyecto, en el cual se comparen los costos reales con el presupuesto del proyecto.

Durante la fase de terminación, el gerente del proyecto debe preparar por escrito una evaluación del desempeño de cada miembro del equipo y mencionar cómo cada uno ha ampliado sus conocimientos como resultado de su tarea, así como las áreas que es necesario desarrollar más. Si dentro de la estructura organizacional de la compañía, un miembro del equipo del proyecto no depende directamente del gerente, éste debe proporcionar una copia de la evaluación del desempeño al supervisor inmediato de la persona.

Por último, ningún proyecto exitoso debe terminarse sin algún tipo de celebración. Esto puede ir desde una fiesta informal con pizzas después del trabajo, hasta un acontecimiento más formal, con oradores de la organización del cliente y premios o certificados de reconocimiento a los participantes.

Otra actividad importante durante la fase de terminación es realizar reuniones de evaluación posteriores a la conclusión del proyecto. Estas reuniones se deben hacer internamente, dentro de la organización que realizó el proyecto y con el cliente. El propósito de estas reuniones es evaluar el desempeño del proyecto, determinar si se lograron en realidad los beneficios previstos e identificar lo que se puede hacer para mejorar el desempeño en futuros proyectos.

# Capítulo 2

## Teoría Básica

### 2.1 FUNDAMENTOS DE LA PROGRAMACIÓN ESTRUCTURADA

El creciente empleo de las computadoras ha conducido a buscar un abaratamiento del desarrollo de software, paralelo a la reducción del costo del hardware obtenido gracias a los avances tecnológicos. Los altos costos del mantenimiento de las aplicaciones de producción normal también han urgido la necesidad de mejorar la productividad del personal de programación.

En la década del sesenta salieron a la luz pública los principios de lo que más tarde se llamo Programación Estructurada, posteriormente se libero el conjunto de las llamadas "Técnicas para mejoramiento de la Productividad en Programación" (Improved Programming Technologies IPT's), siendo la Programación Estructurada una de ellas.

Los programas pueden ser escritos con un alto grado de estructuración, lo cual les permite ser mas fácilmente comprensibles en actividades tales como pruebas, mantenimiento y modificación de los mismos. Mediante la programación Estructurada todas las bifurcaciones de control de un programa se encuentran estandarizadas, de forma tal que es posible leer la codificación del mismo desde su inicio hasta su terminación en forma continua, sin tener que saltar de un lugar a otro del programa, siguiendo el rastro de la lógica establecida por el programador.

Con esta técnica los programadores deben profundizar al proceder realizar el diseño original del programa, el resultado final es más fácil de leer y comprender, el objetivo de

un programador profesional al escribir programas de una manera estructurada, es realizarlos utilizando solamente un número de bifurcaciones de control estandarizados.

El resultado de aplicar la disciplina en la elaboración de programas establecida por la Programación Estructurada, es una programación de alta precisión como nunca antes había sido lograda. Las pruebas de los sistemas, desarrollados utilizando este método, se acoplan más rápidamente y el resultado final es que pueden ser leídos, mantenidos y modificados por otros programadores con mucho mayor facilidad.

Ésta es una técnica en la que se hace uso extensivo de tres estructuras lógicas de control:

- *Secuencia.* Sucesión simple de dos o más operaciones
- *Selección.* Bifurcación condicional de una o más operaciones
- *Interacción.* Repetición de una operación mientras se cumple una condición

Estos tres tipos de estructuras lógicas de control pueden ser combinadas para producir programas que manejen cualquier tarea de procesamiento de información.

Un programa estructurado está compuesto de segmentos, los cuales pueden estar constituidos por unas pocas instrucciones, por una página o más de codificación. Cada segmento tiene solamente una entrada y una salida, estos segmentos, asumiendo que no poseen lazos infinitos y no tienen instrucciones que jamás se ejecuten, se denominan programas propios.

Cuando varios programas propios se combinan utilizando las tres estructuras básicas de control mencionadas anteriormente, el resultado es también un programa propio.

Esta técnica está basada en el Teorema de la Estructura, el cual establece que cualquier programa propio (un programa con una entrada y una salida exclusivamente) es equivalente a un programa que contiene solamente las estructuras lógicas mencionadas anteriormente.

Una característica importante en un programa estructurado es que puede ser leído en secuencia, desde el comienzo hasta el final sin perder la continuidad de la tarea que cumple el programa, lo contrario de lo que ocurre con otros estilos de programación.

Esto es importante debido a que, es mucho más fácil comprender completamente el trabajo que realiza una función determinada, si todas las instrucciones que influyen en su acción están físicamente contiguas y encerradas por un bloque.

La facilidad de lectura, de comienzo a fin, es una consecuencia de utilizar solamente tres estructuras de control y de eliminar la instrucción de desvío de flujo de control, excepto en circunstancias muy especiales tales como la simulación de una estructura lógica de control en un lenguaje de programación que no la posea.

### **Ventajas potenciales**

Un programa escrito de acuerdo a estos principios no solamente tendrá una estructura, sino también una excelente presentación.

Un programa escrito de esta forma tiende a ser mucho más fácil de comprender que los programas escritos en otros estilos. Es fácil para leer y el cual está compuesto de segmentos bien definidos, tiende a ser simple, rápido y menos expuesto a mantenimiento.

La facilidad de comprensión del contenido de un programa puede facilitar la revisión de la codificación y reducir el tiempo de prueba y depuración.

La programación estructurada ofrece estos beneficios, pero no se la debe considerar la panacea ya que el desarrollo de programas es, principalmente, una tarea de dedicación, esfuerzo y creatividad.

## Teorema de la estructura

El teorema de la estructura establece que un programa propio puede ser escrito utilizando solamente las siguientes lógicas de control: secuencia, selección e iteración.

Un programa se define como propio si cumple con los dos requerimientos siguientes:

- Tiene exactamente una entrada y una salida para control del programa.
- Existen caminos desde la entrada hasta la salida que conducen por cada parte del programa, es decir, no existen lazos infinitos ni instrucciones que no se ejecutan.

Las tres estructuras lógicas de control básicas, se definen de la siguiente forma:

*Secuencia*: es simplemente la formalización de la idea de que las instrucciones de un programa son ejecutadas en el mismo orden en que ellas aparecen en el programa. En términos de diagrama de flujo la secuencia es representada por una función después de la otra, como se muestra a continuación.



FIGURA 2.1.1 Secuencia

A y B pueden ser instrucciones sencillas hasta módulos completos, lo importante es que sean programas propios, independientes de su tamaño o complejidad interna. A y B deben ser programas propios en el sentido en que estos fueron definidos, es decir, que posean solamente una entrada y una salida; la combinación de A seguida por B es también un programa propio, ya que esta unión tiene una entrada y una salida exclusivamente, esto se muestra gráficamente en la figura siguiente:

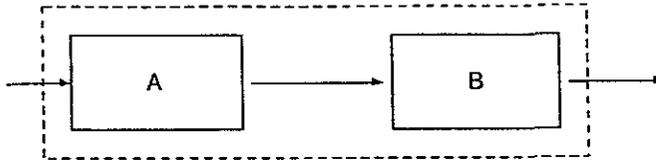


FIGURA 2.1.2 Programa propio

Donde la caja externa sugiere que la combinación de A seguida de B puede ser tratada como una unidad para propósitos de control.

*Selección:* Es la opción entre dos acciones tomando la decisión en base al resultado de evaluar un predicado. Esta estructura de control es denominada usualmente IF-THEN - ELSE. La representación en forma de diagrama de flujo de esta estructura lógica de control se muestra a continuación :

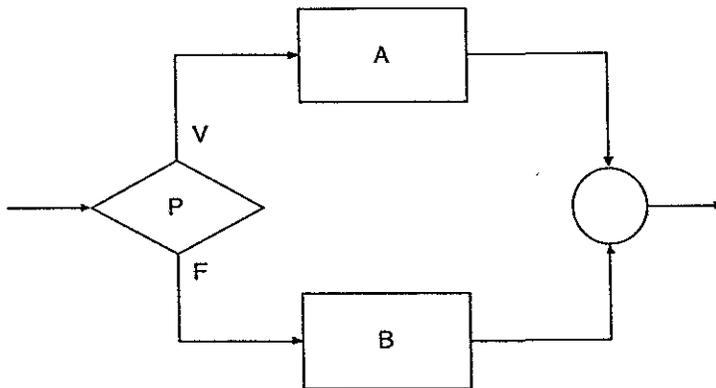


FIGURA 2.1.3 Selección

Donde P es predicado y A y B son las dos funciones.

*Iteración:* Esta estructura lógica es utilizada para que se repita la ejecución de un conjunto de instrucciones mientras se cumpla una condición o predicado. Generalmente a esta estructura se le conoce como DO-WHILE y su representación se muestra a continuación:

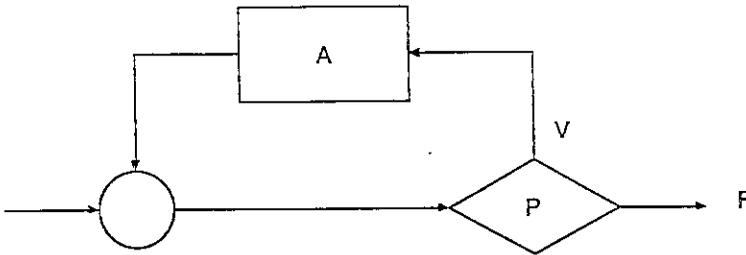


FIGURA 2.1.4 Iteración

donde P es predicado y A es el módulo controlado.

Se debe comprender claramente que un rectángulo, que representa un módulo en un diagrama, siempre puede ser sustituido por cualquiera de las tres estructuras de control descritas anteriormente; por ejemplo, veamos el diagrama siguiente:

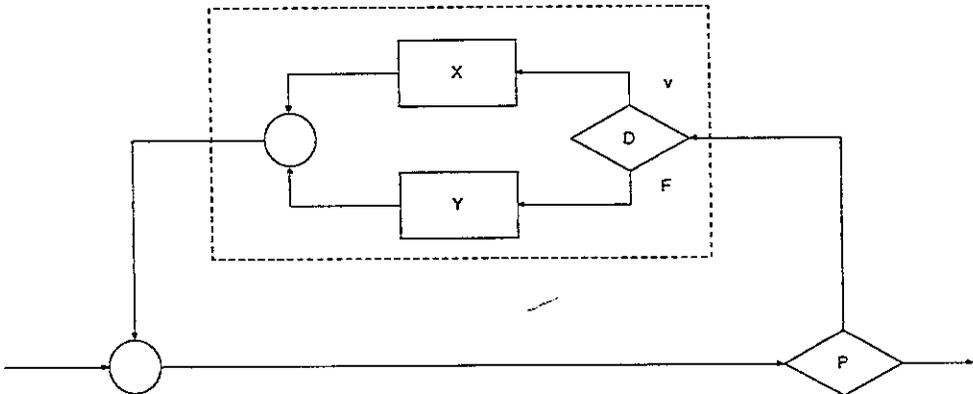


FIGURA 2.1.5 Módulo

En él, la línea punteada limita un rectángulo que contiene una estructura, que a su vez controla dos módulos X y Y. La estructura limitada por la línea punteada es sustituida por una función quedando de la siguiente forma:

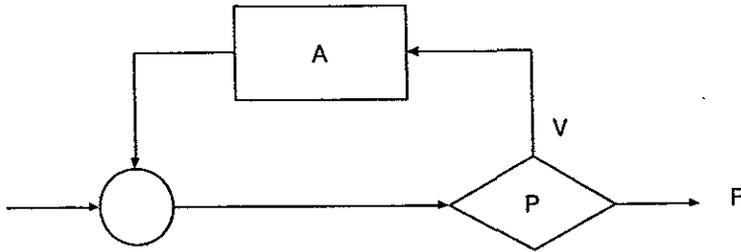


FIGURA 2.1.6 Función

es decir, una función sustituye a una estructura lógica de control o viceversa.

### Otras estructuras lógicas de control

Aunque todos los programas pueden ser escritos utilizando solamente las tres estructuras de control descritas anteriormente es algunas veces, conveniente utilizar algunas estructuras adicionales; a continuación se hará una descripción de esas formas lógicas de control diferentes a la ya definidas:

**DO-UNTIL:** Las estructuras de la iteración básica es el DO-WHILE, pero existe una estructura que es muy parecida a ella y que a veces es usada, dependiendo del procesos que se este tratando de representar y de las características apropiadas en el lenguaje con el cual se esta trabajando, esta forma de control es la que se llama DO-UNTIL, cuya representación gráfica en forma de diagrama de flujo se muestra a continuación:

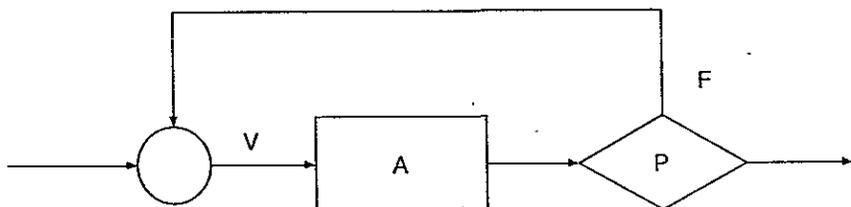


FIGURA 2.1.7 3 Iteración DO - UNTIL

Donde A es el modulo y P el predicado.

La diferencia entre el DO - WHILE y el DO - UNTIL es que en el primero el predicado es probado antes de ejecutar la función, si el predicado es falso la función no es ejecutada; mientras que en el segundo, el predicado es probado después de ejecutar la función, o sea, que la función siempre será ejecutada al menos una vez, independientemente si el predicado es cierto o falso.

*La estructura CASE:* Algunas veces resulta de gran ayuda, desde el punto de vista de eficiencias y facilidad de lectura de un programa, tener alguna forma de expresar una desviación del flujo de control hacia múltiples procesos en función del resultado de la evaluación de un predicado; usualmente, a la estructura de control que satisface el requerimiento anterior, se le denomina la estructura CASE. Por ejemplo, si es necesario ejecutar una de cien rutinas diferentes en función del valor de un código de 2 dígitos, podemos representar este proceso mediante 100 estructuras IF, sin embargo el sentido común nos induce a pensar que no hay razón para adherimos rígidamente a las tres estructuras básicas de control y en lugar de 100 IF usaríamos la estructura CASE.

Esa estructura utiliza el valor de una variable para determinar cual, de varias rutinas, será ejecutada.

En cada lenguaje será necesario establecer cuales son las instrucciones que, en forma conveniente y eficiente, realizan funciones establecidas por las diferentes estructuras lógicas de control.

### **Etiquetas e instrucción GOTO**

Ocasionalmente se habla de la programación estructurada como una técnica de programación que no utiliza GOTO (instrucción de desvío del flujo de control en forma incondicional); si bien es cierto que un programa bien estructurado tiene, o bien

ninguna o muy pocas instrucciones GOTO, asumiendo que estamos empleando un lenguaje de programación adecuado, la ausencia de instrucciones GOTO puede ser mal interpretada. Es conveniente que aclaremos este aspecto en este momento.

Un programa bien estructurado gana una parte importante de su fácil comprensión del hecho que puede ser leído en forma secuencial sin desvíos en el flujo de control desde una parte del programa a otra. Esta característica es consecuencia de usar exclusivamente las estructuras lógicas de control estándar (el GOTO no es una de ellas), esta secuencialidad o lectura TOP\_DOWN es beneficiosa debido a que hay un límite definido para muchos detalles que la mente humana puede abarcar de una vez. Se hace relativamente fácil y rápida la comprensión de la tarea que realiza una instrucción si su función puede ser entendida en términos de una pocas instrucciones mas, físicamente contiguas y delimitadas.

El problema con la instrucción GOTO es que generalmente aleja al programa realizado de los propósitos descritos y en casos extremos puede hacer que un programa sea esencialmente incomprensible.

No se requieren esfuerzos especiales para eliminar de un programa los GOTO, los cuales han sido, algunas veces, malentendidos como enemigos de la programación estructurada, existen buenas y fundadas razones para no querer usuarios pero no se necesita que se realice un trabajo arduo para eliminarlo; ellos no aparecerán, en general, cuando se utilicen las estructuras lógicas de control, descritas anteriormente. Naturalmente, si escogemos para programar un lenguaje de computación que no posea las estructuras lógicas de control fundamentales, entonces, tendremos que simularlas y seguramente ello implicará el uso de la instrucción GOTO; pero este puede hacerse en forma cuidadosa y controlada.

Existen situaciones poco comunes en las cuales el uso de GOTO puede tener ventajas comparado con otras maneras de expresar un proceso; estos casos son excepcionales y usualmente no ocurren en la programación realizada diariamente.

Se deben analizar cuidadosamente las consecuencias de emplear el GOTO, antes de su uso.

## **Segmentación**

Para que la comprensión de un programa se haga de forma fácil y rápida es necesario que, al revisarlo, uno no tenga que hojear una gran cantidad de paginas para entender cual es el trabajo que realiza. Una regla practica para lograr estos fines es establecer que cada segmento del programa no exceda, en longitud a una pagina de codificación, o sea, alrededor de 50 líneas.

La segmentación no es solamente particionar un programa en trozos cuya longitud sea de 50 líneas; esta técnica debe cumplir con ciertas características fundamentales:

- La segmentación reflejara la división del programa en partes que se relacionen entre sí en forma jerárquica, formando una estructura de árbol. Esta organización puede ser representada gráficamente por un diagrama de procesos, lo que hace mas sencillo comprender la relación existente entre un segmento y el resto del programa. Adicionalmente, podemos indicar que, el segmento en la cumbre de la estructura jerárquica contendrá las funciones de control de mas alto nivel, mientras que los segmentos inferiores en esta organización convendrán funciones detalladas.
- Una segmentación bien diseñada deberá mostrar, claramente las relaciones existentes entre las distintas funciones de manera que sea fácil comprender lo que debe hacer el programa y asegurar que efectivamente lo realice. Este hecho, garantizara que los cambios que se efectúen a una parte del programa, durante la programación original o su mantenimiento, no afecten al resto del programa que no ha sufrido cambios.
- En una segmentación bien realizada, la comunicación entre segmentos se lleva a cabo de una manera cuidadosa. Algunos autores recomiendan que los segmentos

consistan en procedimientos y la única comunicación existente entre ellos sea a través de una lista de parámetros, esto reduce la oportunidad de que interactuen entre ellos de una manera indeseada e inentendible.

## Identación

El uso de la identación es importante debido a que, cuando se es consistente en su utilización, facilita la lectura del programa al mostrar en una forma gráfica las relaciones existentes entre las distintas instrucciones.

### Directrices para indentar

Debe comprenderse claramente que las líneas siguientes solo pretenden presentar unas directrices de indentación, sin pretender que estas sean las únicas reglas a seguir en este proceso, cada centro de procesamiento deberá establecer sus propias convenciones, sin que sea motivo de preocupación la diferencia respecto a las sugerencias dadas aquí, lo importante es que se establezcan unas normas y se cumplan de manera consistente.

Las siguientes son sugerencias para el desarrollo de una política de indentación en un centro de procesamiento.

- En los lenguajes donde se permite el uso de etiquetas, estas deben colocarse lo mas externas posibles, por ejemplo comenzando en la columna 2 y deben estar separadas por una línea (siempre que lo permita el lenguaje en uso)
- Se obtiene consistencia si todas las instrucciones se comienzan en una misma columna, por ejemplo en la columna 5 o cualquier otra ubicada a su derecha.
- En los lenguajes en que se hagan declaraciones sobre las variables a utilizar, la información quedara mas claramente representada si los atributos declarados se alinean en forma vertical.

- El uso de líneas en blanco ayuda a mostrar con mas claridad las relaciones existentes entre distintos items agrupados en las declaraciones.
- Las instrucciones son mucho mas fáciles de localizar y de cambiar si no se escribe mas de una instrucción por línea.
- La visión de control de las estructuras lógicas o de los bloques se clarifica si las instrucciones controladas son idénticas por alguna cantidad constante. Se sugiere una indentación de tres espacios.

### **Ventajas de la programación estructurada**

Con la programación estructurada elaborar programas sigue siendo una labor que demanda esfuerzo, creatividad, habilidad y cuidado. Sin embargo, con este estilo podemos obtener las siguientes ventajas:

1. Los programas son más fáciles de entender. Un programa estructurado puede ser leído en secuencia, de arriba hacia abajo, sin necesidad de estar saltando de un sitio a otro en la lógica, lo cual es típico de otros estilos de programación. La estructura del programa es mas clara puesto que las instrucciones están mas ligadas o relacionadas entre sí, por lo que es mas fácil comprender lo que hace cada función.
2. Reducción del esfuerzo en las pruebas. El programa se puede tener listo para producción normal en un tiempo menor del tradicional; por otro lado, el seguimiento de las fallas (debugging) se facilita debido a la lógica mas visible, de tal forma que los errores se pueden detectar y corregir mas fácilmente.
3. Reducción de los costos de mantenimiento
4. Programas más sencillos y más rápidos
5. Aumento de la productividad del programador

6. Se facilita la utilización de las otras técnicas para el mejoramiento de la productividad en programación
7. Los programas quedan mejor documentados internamente.

## **2.2 METODOLOGÍA DE DISEÑO Y ANÁLISIS DE SISTEMAS**

### **Metodología de análisis y diseño**

La metodología es un conjunto de procedimientos, técnicas, herramientas y documentación, para apoyar y facilitar el proceso de diseño y desarrollo de un sistema.

### **Ciclo de Vida de un Sistema de Información**

Conjunto de fases por las que pasa a lo largo del tiempo, desde el estudio del problema, concepción, realización, explotación y mantenimiento. Las fases o etapas son definidas por las diversas metodologías, todas coinciden en una secuencia lógica.

### **Problemas que se presentan en el Desarrollo de Sistemas**

Los proyectos han tenido una serie de problemas debido a la falta de una metodología adecuada para el desarrollo. Las dificultades pueden ser:

1. Sistemas difíciles de administrar.  
En la medida que crecen los sistemas el Administrador pierde el control sobre ellos.  
Sistemas poco confiables
2. Falta de Planeación y optimización
3. Programas poco confiables

Al concluir el desarrollo se presentan resultados inciertos

4. Sistemas difíciles de mantener

Cuando los sistemas son grandes y no existe documentación; los cambios implican tiempo, costo y son difíciles de entender. En ocasiones es mejor volverlos hacer.

5. Sistemas ineficientes

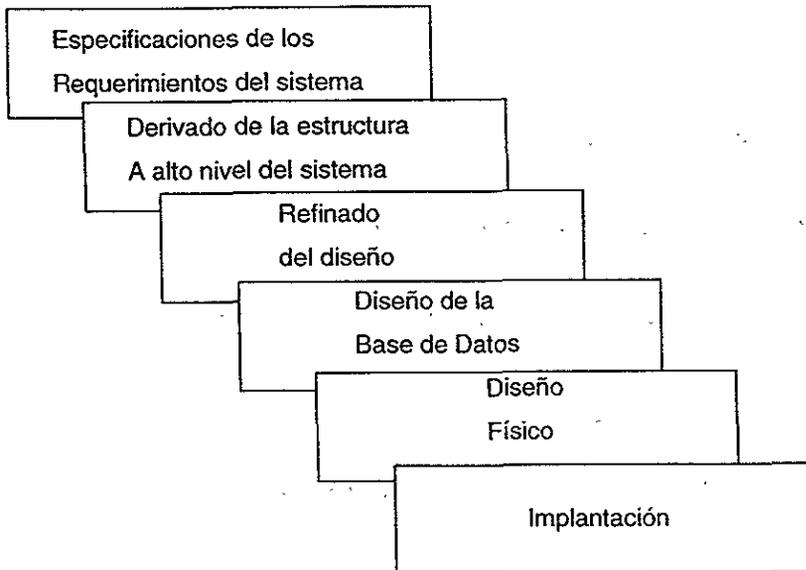
Los sistemas deben de diseñarse para que ocupen la menor cantidad de memoria y el menor tiempo del procesador, además deben de ser los programas claros y modulados.

La construcción de software es un proceso que puede en ocasiones volverse complejo, dependiendo de las características del sistema a ser desarrollado. Por lo que es importante definir enfoques sistemáticos para el desarrollo de software. Cada uno de estos intentos por formalizar los procesos en la construcción de software es denominado metodología de desarrollo.

### **Ingeniería de Software**

La Ingeniería de Software es una disciplina general de desarrollo de Sistemas de Información. Esta propone un proceso de desarrollo paso a paso que comienza con la visión funcional general, descompone esta función en subfunciones y repite el proceso por cada subfunción hasta que estas últimas son codificadas. El resultado es un programa modular jerárquicamente estructurado.

La figura 2.1.1.1 se muestra una visión dinámica del flujo de datos antes de centrarse en una descomposición funcional.



**FIGURA 2.1.1.1 Método utilizado en Ingeniería de Software**

Los tipos de diagramas necesarios para soportar un planteamiento en ingeniería de software para el desarrollo de sistemas son:

- Diagrama de flujo de datos. Muestra los procesos funcionales desde el nivel más alto en un sistema, el flujo de datos, procesos y salidas.
- Diagramas estructurados en árbol. Muestran las relaciones entre los diseños del programa y la estructura jerárquica de los datos.
- Diagrama detallado de la lógica del procedimiento. Detalla la lógica del programa: secuencia, selección e iteración.

La Ingeniería de Software representa uno de los primeros intentos para reestructurar el proceso de desarrollo del software. Desde los años sesenta se ha avanzado en el cambio de desarrollo desde un arte manual a una disciplina casi mecanizada con el objetivo de introducir un enfoque normalizado al desarrollo. Los conceptos y técnicas involucrados son:

- Diseño ascendente
- Programación estructurada
- Modularidad
- Refinamiento progresivo y descomposición funcional
- Métrica de calidad
- La abstracción de los lenguajes de programación.

La metodología de análisis de DeMarco, Gane-Searson y Yourdon son ejemplos de metodologías estructuradas populares pertenecientes a la escuela de Ingeniería de Software.

Dentro de las metodologías de desarrollo, se pueden distinguir:

### **Merise**

Surge en Francia a partir del año 1977, como un intento de definir una metodología a emplear en la Administración Pública para el desarrollo y diseño de sistemas informáticos. Sus principios generales son:

- Desglose del desarrollo por etapas
- Definición de documentos estándar de cada etapa.
- Uso del modelo Entidad - Relación y sus principios para la representación de datos.
- Uso de las redes de Petri para la representación de procesos y tratamientos.
- Definición de grupos de trabajo y reparto de las responsabilidades y funciones a lo largo del desarrollo.
- Especificación del reparto de tareas y tratamientos entre los usuarios y la computadora.
- Definición de los flujos de información entre las unidades del sistema.

La concepción del sistema para un análisis y construcción consiste en trabajar sobre dos elementos: datos y tratamientos sobre los mismos. Los primeros son la información

que se maneja en el entorno del sistema y las relaciones dentro de ella. Los tratamientos reflejan los procesos que se aplican a la información y su secuencia en el tiempo para obtener los resultados deseados. Estos dos elementos son tratados en tres niveles: conceptual, lógico y físico.

La técnica que utiliza Merise es:

- Utilización de grupos de trabajo para el desarrollo del proyecto (comité de directores, comité de usuarios y grupo de desarrolladores).
- Representación gráfica de diagramas de flujo.
- Modelado de datos (conceptual, lógico y físico)
- Modelado de procedimientos (conceptual, lógico y físico)

Las etapas de un desarrollo de un sistema son:

- Estudio Preliminar

Fase 1: Recolección de Datos

Recolección Inicial

Estudio de la Situación Actual

Síntesis y Crítica de la Situación Actual.

Fase 2: Concepción de la nueva solución

Objetivo a alcanzar

Descripción de la solución

Fase 3: Evaluación y plan de desarrollo

Evaluación de la nueva solución

Plan de desarrollo

- Estudio Detallado

Fase 1: Concepción general

Fase 2: Concepción detallada de las fases.

Realización de las especificaciones detalladas de procesos.

### Fase 3: Plan de desarrollo

- Realización
  - Fase 1: Estudio Técnico
  - Fase 2: Producción
  
- Puesta en marcha
  - Fase 1: Preparación de los recursos físicos y humanos
  - Fase 2: Recepción y lanzamiento del sistema.

Para esta metodología utiliza el modelo Entidad - Relación como una representación gráfica con el fin de mostrar entidades con significado dentro del sistema y sus relaciones entre ellas.

Las redes de Petri son otro método de representación acción-operación a realizar determina:

- Qué sucesos desencadenan la ejecución de procesos
- Bajo qué condiciones de esos sucesos desencadenantes se ejecuta el proceso.
- En qué consiste el proceso, es decir, qué operaciones hay que realizar.
- Quién ejecuta el proceso
- Resultados obtenidos o salidas del proceso.

En donde se define "Proceso" como toda actividad de la empresa con sus entradas y salidas establecidas, y "Operación o Procedimiento" como una serie de acciones ejecutables sin interrupción.

### **Metodología de Yourdon**

Esta metodología surge en Estados Unidos con su precursor Edward. Yourdon, los métodos que usa son arriba-abajo y descomposición funcional para definir los requerimientos del sistema.

---

El ciclo de vida de la metodología propone:

- **La encuesta**

También se le conoce como estudio de factibilidad o inicial del negocio. Comienza cuando el usuario solicita que una o más partes de los procesos se automaticen. Entre los objetivos de la encuesta se tienen los siguientes:

1. Identificar a los usuarios responsables y crear un entorno inicial del sistema.
2. Establecer metas y objetivos para el nuevo sistema.
3. Determinar si es factible automatizar el sistema
4. Preparar el esquema que se usará para guiar el resto del proyecto.

- **Análisis Estructurado**

El propósito principal de la actividad de análisis es transformar la política del negocio y el esquema del proyecto, en una especificación estructurada; empleando herramientas de diseño como son los diagramas de flujo, diagrama entidad-relación, etc.

- **Diseño**

Crea una jerarquía de módulos de programa e interfaces, para implementar la especificación creada durante el análisis, se transforma el modelo entidad-relación en un diseño de Base de Datos.

- **Codificación**

Programación e integración de los módulos, aquí entra la etapa de la metodología de programación como es de arriba-abajo, de lo general a lo particular.

- **Elaboración de pruebas de aceptación.**

Producción de un conjunto de pruebas para la aceptación del usuario, por medio de especificaciones ya definidas.

- **Garantía de Calidad**

El sistema debe de tener un tiempo de respuesta aceptable, codificación entendible en cada una de las fases para asegurar la calidad del producto.

- **Elaboración de Manuales de usuario**

Este manual contendrá la descripción de la interacción del usuario con el Sistema.

- **Conversión de los datos del Sistema anterior**

Si existe un sistema anterior, se debe de convertir la información actual al nuevo Sistema, esto implica la realización de programas u otro Sistema; dependiendo de la complejidad y cantidad de la información.

- **Instalación**

Comprende desde capacitación al usuario, entrega de manuales, procesos en paralelo en el caso de existir otro sistema (no es recomendable).

A continuación se muestra en la figura 2.1.1.3 las Fases de Desarrollo según Yourdon

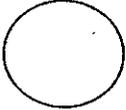
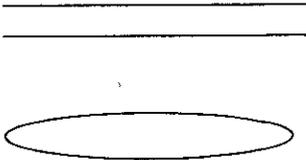


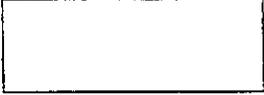
FIGURA 2.1.1.3 Fases de Desarrollo según Yourdon

Las siguientes herramientas de modelado ayudarán a concentrarse en las propiedades más sobresalientes del sistema, aclarar cambios y correcciones de los requerimientos de los usuarios.

### Diagrama de Flujo

Describe los procesos, es decir, la transformación de entradas y salidas. Esta es una herramienta que permite visualizar un sistema como una red de procesos funcionales, conectados entre sí por líneas de flujo y almacenamiento de datos, estos diagramas proporcionan un punto de vista de un sistema, el orientado a las funciones. Sus componentes se muestran en la figura 2.1.1.4:

<p><b>Procesos.</b> Son las diferentes funciones que el sistema lleva a cabo y es representado por un círculo. El nombre se asigna dependiendo de lo que este efectuando.</p>	
<p><b>Flujos.</b> Son las conexiones entre procesos, e indica la información que se requiere como entrada y la información que genera como salida de cada proceso, representan datos en movimiento. Se representan por medio de una flecha que entra o sale del proceso. Pueden ser acompañados por un nombre señalando el significado del paquete de información que se mueve a lo largo del flujo.</p>	
<p><b>Agregados o almacén de datos.</b> Son colecciones de datos que el sistema guarda para ser llamados en un período de tiempo, es decir, representan las entidades en el modelado lógico o los datos en reposo. Dependiendo de la herramienta utilizada se denota por líneas paralelas o elipses.</p>	

<p><b>Terminadores.</b> Muestran las entidades externas con las que se comunicará el sistema, por ejemplo organizaciones externas, sistemas de cómputo, etc. Se representa por un rectángulo.</p>	
---	--

**FIGURA 2.1.1.4 Diagrama de Flujo de Datos**

Para mostrar con más detalle el diagrama de flujo de datos se hace uso de dos herramientas de modelado: Diccionario de datos y especificaciones de proceso.

Algunas de las reglas que ayudan a elaborar un DFD con el mínimo de errores y cambios son:

- Elegir nombres con significado para los procesos, flujos, almacenes y terminadores.
- Enumerar los procesos.
- Evitar DFD complejos.
- Asegurarse que el DFD sea internamente consistente y con cualquier DFD relacionado.

### Diagrama Entidad-Relación

Los diagramas de flujo de datos sólo muestran el aspecto de sus funciones, para conocer más detalle la relación que existe entre las entidades se utiliza esta herramienta que enfatiza las relaciones entre los almacenes de datos que se describen en el Diagrama de Flujo.

El diagrama entidad-relación sirve para relacionar entidades, ésta puede ser “uno a uno” o “muchos a muchos”, etc. En el diagrama de la figura 2.1.1.5 muestra las relaciones y entidades.

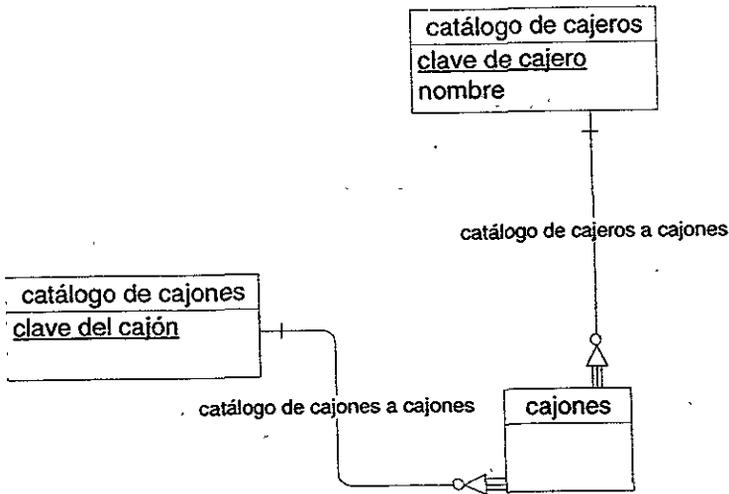


FIGURA 2.1.1.5 Diagrama Entidad-Relación

El diagrama Entidad-Relación es acompañado de información textual por medio del diccionario de datos, que mantiene información apropiada acerca de los objetos y relaciones.

### Diagrama de Transición de Estados

Es la secuencia con la cual se tendrá acceso a los datos y se ejecutarán las funciones, por ejemplo la función N no puede llevarse a cabo hasta que recibe la entrada que requiere; y esta entrada se produce como salida de una función N-1, y así sucesivamente.

Los estados en los que se puede presentar el sistema ("situaciones reconocibles"), representan un período durante el cual, el sistema sigue algún comportamiento, cualquier estado en el que el sistema se pueda encontrar sólo puede corresponder a períodos en los que está esperando que algo ocurra en el ambiente externo o está esperando a que alguna actividad presente en ese momento cambie a otro estado.

Existen acciones asociadas con cada cambio de estado, esto es, respuestas, salidas o actividades que se llevan a cabo como parte del cambio de estado. Para realizar estos cambios de estado será necesario establecer una serie de condiciones en el sistema.

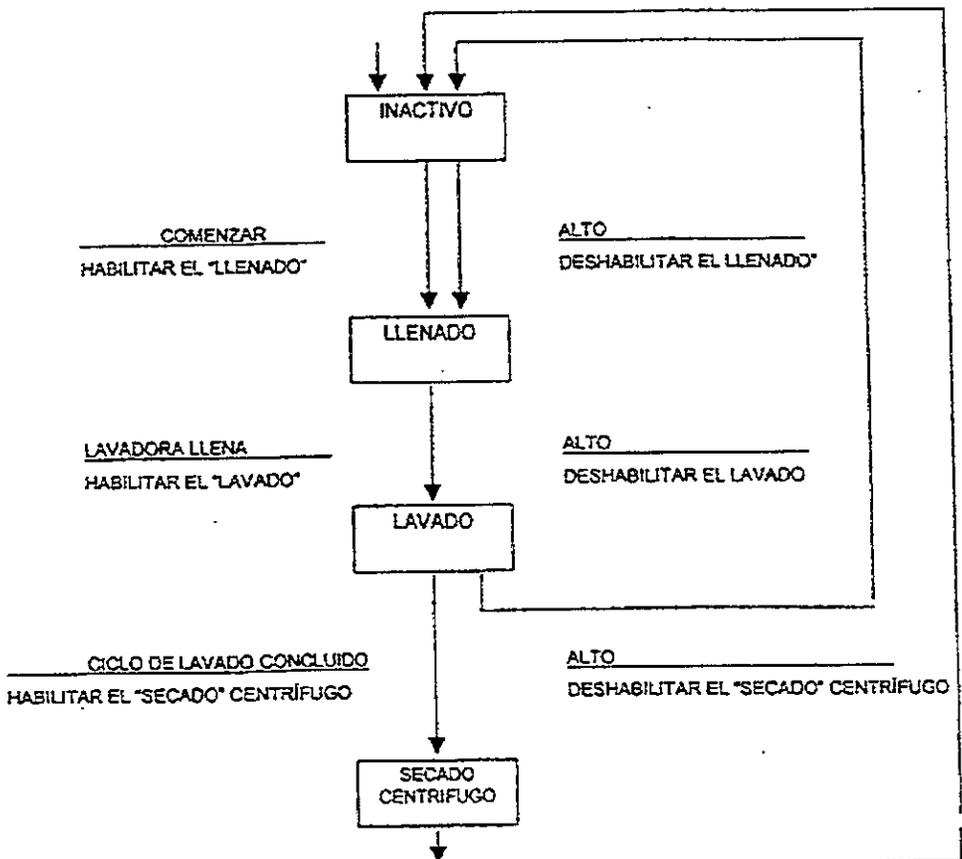


FIGURA 2.1.1.6 Diagrama de Transición de Estados

Una condición es un acontecimiento en el ambiente externo que el sistema es capaz de detectar, puede ser una señal, interrupción a la llegada de un paquete de estados.

Las acciones se muestran en un Diagrama de Transición de Estados(DTE) son respuestas regresadas al ambiente externo o bien cálculos cuyos resultados el sistema recuerda para poder responder a algún acontecimiento futuro.

### Diagrama de Estructuras

Son herramientas que emplean los diseñadores de sistemas y representan la jerarquía de Software. En la figura 2.1.1.7 se muestra un diagrama típico, en donde cada rectángulo representa las invocaciones de los módulos (llamados de subrutinas o procedimientos), así como los parámetros de entrada que se dan a cada módulo invocado, y los parámetros de salida devueltos por cada módulo cuando termina su labor y le devuelve el control al que lo llama.

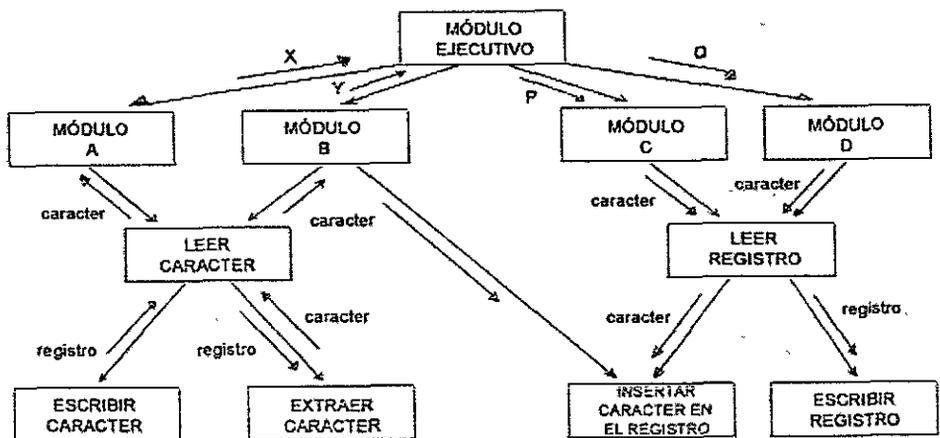


FIGURA 2.1.1.7 Diagrama de Estructura.

## Metodología SSADM

SSADM (Structured System Analysis and Design Method), nace en el Reino Unido a petición del Gobierno Inglés con la intención de ser un sistema de desarrollo público de Gran Bretaña.

Es aceptada en 1981 como primer versión. Se hace uso obligatorio en toda la administración en 1983, hasta el modelo actual.

La metodología consiste en una estructuración de pasos a seguir, la técnica y formalismos de cada fase. Se puede ver de una manera jerárquica como se muestra en la figura 2.1.1.8

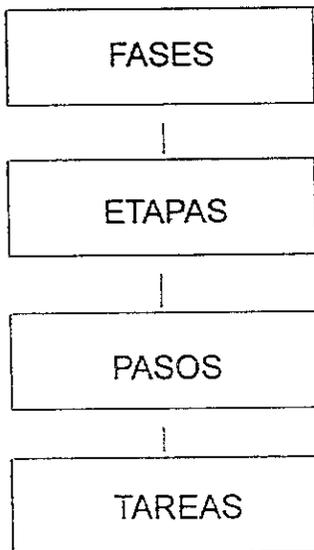


FIGURA 2.1.1.8 Estructuración de pasos en un proyecto informático

El método contempla, las tres primeras fases del desarrollo (estudio de viabilidad, análisis y diseño) divididas a su vez en una serie de etapas, como se muestra en la figura 2.1.1.9

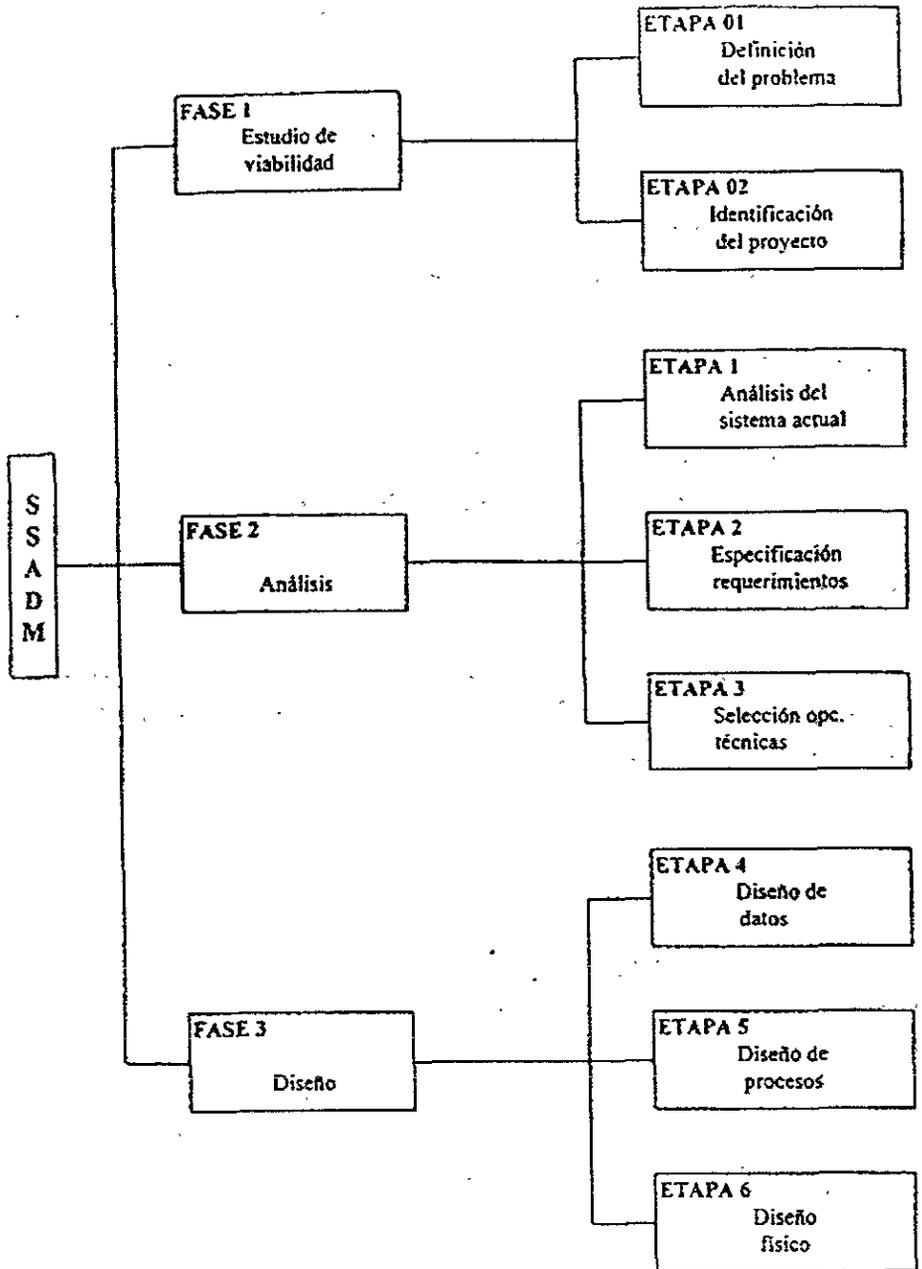


FIGURA 2.1.9 Fases y etapas de metodología SSADM

Una vez comprobada la viabilidad del proyecto las etapas se desarrollan en el siguiente orden.

Las técnicas utilizadas en cada etapa son:

- Diagrama de Flujo de Datos, representación de los flujos de información y sus relaciones con otros sistemas.
- Estructura Lógica de Datos, emplea modelo entidad-relación
- Historia en la Vida de la Entidad (ELH), representa la descripción de cómo las entidades descritas son afectadas por diferentes sucesos que ocurren en el sistema.
- Tercera Forma Normal, es el método matemático para la definición de datos que ayuda a evitar inconsistencias y ambigüedades en la información.

## **Conclusión**

Para cada una de las metodologías antes mencionadas existe puntos comunes y diferentes, la mejor metodología es la que se adapte al negocio en base a las etapas del ciclo de vida, modelado de datos y procesos.

## **2.3 SISTEMAS OPERATIVOS**

Sin el software, una computadora no es más que una masa metálica sin utilidad. Con el software, una computadora puede almacenar, procesar y recuperar información, encontrar errores de ortografía en manuscritos, tener aventuras e intervenir en muchas otras valiosas actividades para ganar el sustento. El software para computadoras puede clasificarse en general en dos clases: los programas de sistema, que controlan la operación de la computadora en si y los programas de aplicación, los cuales resuelven problemas para sus usuarios.

El programa fundamental de todos los programas de sistema es el sistema operativo (SO), que controla todos los recursos de la computadora y proporciona la base sobre la cual pueden escribirse los programas de aplicación.

### 2.3.1 ¿Qué es un Sistema Operativo?

Desde su creación, las computadoras digitales han utilizado un sistema de codificación de instrucciones en sistema de numeración binaria, es decir con los 0S. Esto se debe a que los circuitos integrados funcionan con este principio, es decir, hay corriente o no hay corriente.

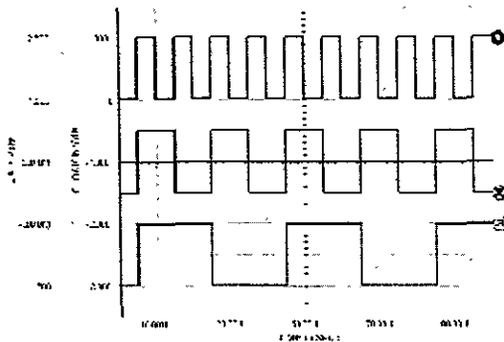


FIGURA 2.3.1.1 Pulsos de corriente eléctrica

En el origen de la historia de las computadoras (hace unos cuarenta años), los sistemas operativos no existían y la introducción de un programa para ser ejecutado se convertía en un increíble esfuerzo que solo podía ser llevado a cabo por muy pocos expertos.

Esto hacía que las computadoras fueran muy complicadas de usar y que se requiriera tener altos conocimientos técnicos para operarlas. Era tan complejo su manejo, que en algunos casos el resultado llegaba a ser desastroso.

Además, el tiempo requerido para introducir un programa en aquellas grandes máquinas de lento proceso superaba por mucho el de ejecución y resultaba poco provechosa la utilización de computadoras para resolución de problemas prácticos.

Se buscaron medios más elaborados para manipular la computadora, pero que a su vez simplificaran la labor del operador o el usuario. Es entonces cuando surge la idea de crear un medio para que el usuario pueda operar la computadora con un entorno, lenguaje y operación bien definido para hacer un verdadero uso y explotación de esta.

Surgen los sistemas operativos. Un sistema operativo es el encargado de brindar al usuario una forma amigable y sencilla de operar, interpretar, codificar y emitir las ordenes al procesador central para que este realice las tareas necesarias y específicas para completar una orden. El sistema operativo, es el instrumento indispensable para hacer de la computadora un objeto útil. Bajo este nombre se agrupan todos aquellos programas que permiten a los usuarios la utilización de este enredo de cables y circuitos, que de otra manera serían difíciles de controlar. Un sistema operativo se define como un conjunto de procedimientos manuales y automáticos, que permiten a un grupo de usuarios compartir una instalación de computadora eficazmente.

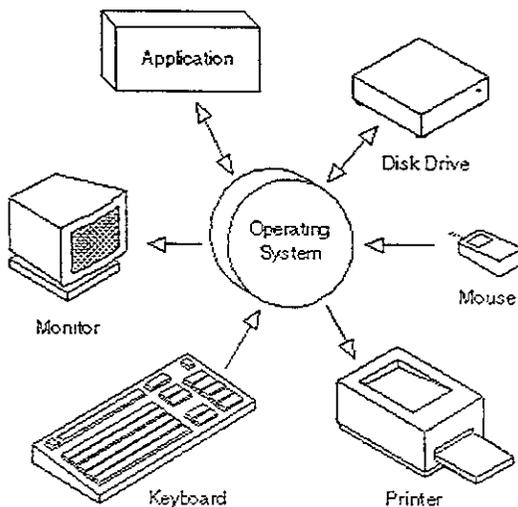


FIGURA 2.3.1.2 Gestión del sistema operativo

### 2.3.2 Interfaz de Línea de Comandos

La forma de interfaz entre el sistema operativo y el usuario en la que este escribe los comandos utilizando un lenguaje de comandos especial. Los sistemas con interfaces de líneas de comandos se consideran más difíciles de aprender y utilizar que los de las interfaces gráficas.

Sin embargo, los sistemas basados en comandos son por lo general programables, lo que les otorga una flexibilidad que no tienen los sistemas basados en gráficos carentes de una interfaz de programación.

### 2.3.3 Interfaz Gráfica del Usuario

Es el tipo de visualización que permite al usuario elegir comandos, iniciar programas y ver listas de archivos y otras opciones utilizando las representaciones visuales (iconos) y las listas de elementos del menú. Las selecciones pueden activarse bien a través del teclado o con el mouse.

Para los autores de aplicaciones, las interfaces gráficas de usuario ofrecen un entorno que se encarga de la comunicación con el ordenador o computadora. Esto hace que el programador pueda concentrarse en la funcionalidad, ya que no está sujeto a los detalles de la visualización ni a la entrada a través del mouse o el teclado. También permite a los programadores crear programas que realicen de la misma forma las tareas más frecuentes, como guardar un archivo, porque la interfaz proporciona mecanismos estándar de control como ventanas y cuadros de diálogo.

Otra ventaja es que las aplicaciones escritas para una interfaz gráfica de usuario son independientes de los dispositivos: a medida que la interfaz cambia para permitir el uso de nuevos dispositivos de entrada y salida, como un monitor de pantalla grande o un dispositivo óptico de almacenamiento, las aplicaciones pueden utilizarlos sin necesidad de cambios.

### 2.3.4 Funciones de los Sistemas Operativos

- Interpreta los comandos que permiten al usuario comunicarse con el ordenador.
- Coordina y manipula el hardware de la computadora, como la memoria, las impresoras, las unidades de disco, el teclado o el mouse.
- Organiza los archivos en diversos dispositivos de almacenamiento, como discos flexibles, discos duros, discos compactos o cintas magnéticas.
- Gestiona los errores de hardware y la pérdida de datos.
- Sirve de base para la creación del software logrando que equipos de marcas distintas funcionen de manera análoga, salvando las diferencias existentes entre ambos.
- Configura el entorno para el uso del software y los periféricos; dependiendo del tipo de máquina que se emplea, debe establecerse en forma lógica la disposición y características del equipo. Como por ejemplo, una microcomputadora tiene físicamente dos unidades de disco, puede simular el uso de otras unidades de disco, que pueden ser virtuales utilizando parte de la memoria principal para tal fin. En caso de estar conectado a una red, el sistema operativo se convierte en la plataforma de trabajo de los usuarios y es este quien controla los elementos o recursos que comparten. De igual forma, provee de protección a la información que almacena.

### 2.3.5 Sistema Operativo Multitareas

Es el modo de funcionamiento disponible en algunos sistemas operativos, mediante el cual una computadora procesa varias tareas al mismo tiempo. Existen varios tipos de multitareas. La conmutación de contextos (context Switching) es un tipo muy simple de multitarea en el que dos o más aplicaciones se cargan al mismo tiempo, pero en el que sólo se esta procesando la aplicación que se encuentra en primer plano (la que ve el usuario). Para activar otra tarea que se encuentre en segundo plano, el usuario debe traer al primer plano la ventana o pantalla que contenga esa aplicación. En la multitarea cooperativa, la que se utiliza en el sistema operativo Macintosh, las tareas en segundo plano reciben tiempo de procesado durante los tiempos muertos de la tarea que se

encuentra en primer plano (por ejemplo, cuando esta aplicación esta esperando información del usuario), y siempre que esta aplicación lo permita. En los sistemas multitarea de tiempo compartido, como OS/2, cada tarea recibe la atención del microprocesador durante una fracción de segundo. Para mantener el sistema en orden, cada tarea recibe un nivel de prioridad o se procesa en orden secuencial. Dado que el sentido temporal del usuario es mucho más lento que la velocidad de procesamiento del ordenador, las operaciones de multitarea en tiempo compartido parecen ser simultáneas.



FIGURA 2.3.5.1 OS/2 Warp, sistema operativo multitarea

### 2.3.6 Sistema Operativo Monotareas

Los sistemas operativos monotareas son más primitivos y es todo lo contrario al visto anteriormente, es decir, solo pueden manejar un proceso en cada momento o que solo puede ejecutar las tareas de una en una. Por ejemplo cuando la computadora esta imprimiendo un documento, no puede iniciar otro proceso ni responder a nuevas instrucciones hasta que se termine la impresión.



FIGURA 2.3.6.1 PalmOs, Sistema Operativo monotareas.

### 2.3.7 Sistema Operativo Monousuario

Los sistemas monousuarios son aquellos que nada más puede atender a un solo usuario, gracias a las limitaciones creadas por el hardware, los programas o el tipo de aplicación que se este ejecutando.

Estos tipos de sistemas son muy simples, porque todos los dispositivos de entrada, salida y control dependen de la tarea que se esta utilizando, esto quiere decir, que las instrucciones que se dan, son procesadas de inmediato; ya que existe un solo usuario. Y están orientados principalmente por los microcomputadores.

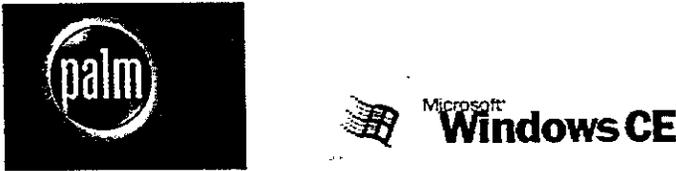


FIGURA 2.3.7.1 Sistemas Operativos monousuarios

### 2.3.8 Sistema Operativo Multiusuario

Es todo lo contrario a monousuario; y en esta categoría se encuentran todos los sistemas que cumplen simultáneamente las necesidades de dos o más usuarios, que comparten los mismos recursos. Este tipo de sistemas se emplean especialmente en redes.

En otras palabras consiste en el fraccionamiento del tiempo (timesharing).



FIGURA 2.3.8.1 Sistemas operativos multiusuarios

---

**ESTA TESIS NO SALE  
DE LA BIBLIOTECA****2.3.9 Secuencia por Lotes**

La secuencia por lotes o procesamiento por lotes en microcomputadoras, es la ejecución de una lista de comandos del sistema operativo uno tras otro sin intervención del usuario. En los ordenadores más grandes el proceso de recogida de programas y de conjuntos de datos de los usuarios, la ejecución de uno o unos pocos cada vez y la entrega de los recursos a los usuarios.

Procesamiento por lotes también puede referirse al proceso de almacenar transacciones durante un cierto lapso antes de su envío a un archivo maestro, por lo general una operación separada que se efectúa durante la noche.

Los sistemas operativos por lotes (batch), en los que los programas eran tratados por grupos (lote) en vez de individualmente. La función de estos sistemas operativos consistía en cargar en memoria un programa de la cinta y ejecutarlo.

Al final este, se realizaba el salto a una dirección de memoria desde donde reasumía el control del sistema operativo que cargaba el siguiente programa y lo ejecutaba. De esta manera el tiempo entre un trabajo y el otro disminuía considerablemente.

**2.3.10 Tiempo Real**

Un sistema operativo en tiempo real procesa las instrucciones recibidas al instante, y una vez que han sido procesadas muestra el resultado. Este tipo tiene relación con los sistemas operativos monousuarios, ya que existe un solo operador y no necesita compartir el procesador entre varias solicitudes.

Su característica principal es dar respuestas rápidas; por ejemplo en un caso de peligro se necesitarían respuestas inmediatas para evitar una catástrofe.

### 2.3.11 Tiempo Compartido

El tiempo compartido en ordenadores o computadoras consiste en el uso de un sistema por más de una persona al mismo tiempo. El tiempo compartido ejecuta programas separados de forma concurrente, intercambiando porciones de tiempo asignadas a cada programa (usuario). En este aspecto, es similar a la capacidad de multitareas que es común en la mayoría de los microordenadores o las microcomputadoras. Sin embargo el tiempo compartido se asocia generalmente con el acceso de varios usuarios a computadoras más grandes y a organizaciones de servicios, mientras que la multitarea relacionada con las microcomputadoras implica la realización de múltiples tareas por un solo usuario.

### 2.3.12 Los Sistemas Operativos más Populares de las PC

Los sistemas operativos empleados normalmente son UNIX, Macintosh OS, OS/2, Windows 95, Windows NT y recientemente LINUX.

#### 2.3.12.1 OS/2

Desarrollado inicialmente por Microsoft Corporation e International Business Machines (IBM), después de que Intel introdujera al mercado su procesador 80286. Pero la sociedad no duro mucho ya que IBM veía a Windows como una amenaza para el OS/2.

Pero IBM continuo desarrollando este sistema operativo. El OS/2 al principio fue muy parecido al MS-DOS, tiene una línea de comando, pero la diferencia que existe con el DOS es el intérprete de comandos, el cual es un programa separado del kernel del sistema operativo y aparece únicamente cuando se hace clic en uno de los iconos "OS/2 prompt" dentro del Workplace Shell.

Otra diferencia es que este sí es un sistema operativo multitarea.

En el OS/2 muchos de los comandos son idénticos a los de su contra parte pero tiene más comandos debido a que es más grande, completo y moderno.

El ambiente gráfico es el Workplace Shell (WS), es el equivalente a un administrador del área de trabajo para el WS.



FIGURA 2.3.12.1.1 Sistema Operativo OS/2 WARP

### 2.3.12.2 Macintosh OS

El sistema operativo constituye la interfaz entre las aplicaciones y el hardware del Macintosh. El administrador de memoria obtiene y libera memoria en forma automática para las aplicaciones y el sistema operativo. Esta memoria se encuentra normalmente en un área llamada cúmulo. El código de procedimientos de una aplicación también ocupa espacio en el cúmulo. Ahora se presenta una lista de los principales componentes del sistema operativo.

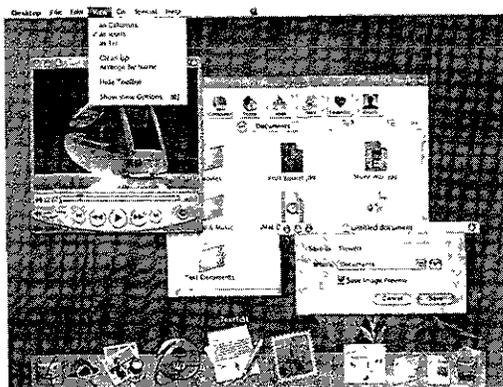


FIGURA 2.3.12.2.1 Escritorio MAC-OS

- El cargador de segmentos carga los programas por ejecutar. Una aplicación se puede cargar completa o bien puede dividirse en segundos individuales que se pueden cargar de manera dinámica conforme se necesiten.
- El administrador de eventos del sistema operativo informa de la ocurrencia de diversos eventos de bajo nivel, como la presión de un botón del mouse o el tecleo. En condiciones normales, el administrador de eventos de la caja de herramientas transfiere estos eventos a las aplicaciones.
- El administrador de archivos se encarga de la entrada / salida de archivos; el administrador de dispositivos se encarga de la entrada / salida de dispositivos.
- Los manejadores de dispositivos son programas con los cuales los diversos tipos de dispositivos pueden presentar interfaces uniformes de entrada / salida a las aplicaciones. Tres manejadores de dispositivo están integrados al sistema operativo en ROM: el manejador de disco se encarga del acceso a la información en discos, el manejador de sonido controla los generadores de sonido, y el manejador en serie envía y recibe datos a través de los puertos seriales (estableciendo así la comunicación con dispositivos periféricos en serie como impresoras y módems).
- Con el manejador de impresoras las aplicaciones pueden imprimir datos en diversas impresoras.
- Con el administrador de AppleTalk las aplicaciones pueden transmitir y recibir información en una red de comunicaciones AppleTalk.
- El Administrador de retrasado vertical programa las actividades por realizar durante las interrupciones de retrasado vertical que ocurren 60 veces cada segundo cuando se refresca la pantalla de vídeo.
- El manejador de errores del sistema toma el control cuando ocurre un error fatal del sistema y exhibe un cuadro de error apropiado.
- Los programas de utilidad general del sistema operativo ofrecen diversas funciones útiles como la obtención de la fecha y la hora, la comparación de cadenas de caracteres y muchas más.
- El paquete de iniciación es llamado por el paquete de archivos estándar para iniciar y nombrar discos; se aplica con más frecuencia cuando el usuario inserta un disco al que no se le han asignado valores iniciales.

- El paquete de aritmética de punto flotante ofrece aritmética de doble precisión. El paquete de funciones trascendentales ofrece un generador de números aleatorios, así como funciones trigonométricas, logarítmicas, exponenciales y financieras. Los compiladores de Macintosh generan en forma automática llamadas a estos paquetes para realizar manipulaciones numéricas.

### 2.3.12.3 UNIX

Es un sistema operativo multiusuario que incorpora multitarea. Fue desarrollado originalmente por Ken Thompson y Dennis Ritchie en los laboratorios de AT&T Bell en 1969 para su uso en minicomputadoras. El sistema operativo UNIX tiene diversas variantes y se considera potente, más transportable e independiente de equipos concretos que otros sistemas operativos porque está escrito en lenguaje C. El UNIX está disponible en varias formas, entre las que se cuenta AIX, una versión de UNIX adaptada por IBM (para su uso en estaciones de trabajo basadas en RISC), A/ux (versión gráfica para equipos Apple Macintosh) y Mach (un sistema operativo reescrito, pero esencialmente compatible con UNIX, para las computadoras NeXT).

El UNIX y sus clones permiten múltiples tareas y múltiples usuarios. Su sistema de archivos proporciona un método sencillo de organizar archivos y permite la protección de archivos. Sin embargo, las instrucciones del UNIX no son intuitivas.



FIGURA 2.3.12.3.1 Sun Solaris, sistema operativo UNIX

Este sistema ofrece una serie de utilerías muy interesantes, como las siguientes:

- Inclusión de compiladores e intérpretes de lenguaje.

- Existencia de programas de interface con el usuario, como ventanas, menús, etc.
- Muchas facilidades a la hora de organización de archivos.
- Inclusión de lenguajes de interrogación.
- Facilidades gráficas.
- Programas de edición de textos.

#### 2.3.12.4 Microsoft Windows NT

Microsoft no solo se ha dedicado a escribir software para PCs de escritorio sino también para poderosas estaciones de trabajo y servidores de red y bases de datos.

El sistema operativo Windows NT de Microsoft, lanzado al mercado el 24 de Mayo de 1993, es un SO para redes que brinda poder, velocidad y nuevas características; además de las características tradicionales. Es un SO de 32 bits, y que puede trabajar en procesadores 386, 486 y Pentium.



FIGURA 2.3.12.4.1 Windows NT de la familia de Microsoft

Además de ser multitarea, multilectura y multiprocesador ofrece una interfaz gráfica. Y trae todo el software necesario para trabajar en redes, permitiendo ser un cliente de la red o un servidor. Es el sistema mas desarrollado de la familia Windows y fue diseñado para poder aprovechar al máximo los recursos del complejo hardware de 32 bits que existe actualmente.

#### 2.3.12.5 Características

- Full 32 bits
- Sistema Operativo Multitarea

- Incluye procesadores INTEL RISC y Sistemas de Multiprocesamientos Simétricos
- Diseño de un avanzado microkernel integrando seguridad y una plataforma robusta para su manejo
- Recursos compartidos de archivos e impresoras
- Trabajo en forma de grupo
- Soporte para múltiples protocolos
- Compatible con otras redes (Banyan, VINES y Novell Netware)
- Manejo centralizado de restricciones a usuarios, a través de una sola computadora, de departamentos, divisiones y grupos
- Protección avanzada de datos a través de discos espejados, segmentación de discos (RAID 5) y soporte para manejo de UPS
- Servicio de acceso remoto que permite conectarse vía WAN a través del discado en líneas telefónicas asincrónicas, ISDN y redes X-25. Estas tecnologías deben aplicarse con computadoras trabajando con MS-DOS, Windows para trabajo en grupo o Sistema Operativo Windows NT
- Servicio para Macintosh (los usuarios de este sistema pueden acceder al Windows NT y compartir aplicaciones con otros servers de APPLÉ)
- Puede trabajar como Sistema Operativo Distribuido aplicando el concepto de Dominio pudiendo ser Server primario o en su defecto secundario.

#### 2.3.12.6 Requerimientos de hardware

- Microprocesadores 32 bits tales como:
- INTEL PENTIUM II a 433 MHz o superior, microprocesador RISC Mips R4000 y DEC Alfa.
- Vídeo: VGA o alta resolución.
- Discos Rígidos: Mínimo 90 Megabytes en el caso de Intel y 110 en caso de sistemas RISC.
- Memoria: 16 Mega de RAM para sistemas x86 y RISC.

Windows NT tiene un soporte para microprocesadores adicionales partiendo del equipamiento original.

### 2.3.12.7 Servicios

Cada grupo de servicios es manejado por uno de los siguientes componentes separados.

- Manejador de objetos
- Manejador de Memoria Virtual
- Manejador de Procesos
- Facilidad de llamadas a procedimientos locales(LPCF)
- Manejador de E/S
- Monitor de referencia de seguridad, que conjuntamente con el Logon and Security, subsistemas protegidos, conforman el modelo de seguridad de Windows NT

#### *MANEJADOR DE OBJETOS:*

Los objetos son instancias en tiempo de ejecución de un objeto tipo particular que puede ser manipulado por un proceso de S.O. Un objeto tipo incluye un tipo de datos definido por sistema, una lista de operaciones que pueden realizarse sobre ellos y un conjunto de objetos de atributos.

El Manejador de objetos es la parte del S.O. (Windows NT) ejecutivo que provee reglas uniformes para retención, nombramiento y seguridad de objetos.

#### *MANEJADOR DE PROCESOS:*

El Manejador de procesos es el componente que rastrea dos tipos de objetos (objetos procesos y objetos hilo). Un proceso se define como un espacio de direcciones, un conjunto de objetos (recursos) visibles al procesos y un conjunto de hilos que se ejecutan en el contexto del proceso. Un hilo es la entidad más básica planificable del sistema. Tiene su propio conjunto de registros, su propia pila de Kernel, un bloque de ambientes de hilos y una pila de usuario en el espacio de direcciones de ese proceso.

*El Manejador de procesos es el componente que maneja la creación y eliminación de procesos. Provee un conjunto de servicios estándar para la creación y uso de hilos y procesos en el contexto de un ambiente de un subsistema particular. El Manejador de procesos no impone ninguna jerarquía o grupo de reglas para los procesos ni fuerza ninguna relación padre/hijo.*

El modulo de proceso de Windows NT trabaja en conjunción con el modelo de seguridad y con el Manejador de Memoria Virtual para proveer protección a los interprocesos. A cada proceso se le asigna un token de seguridad del acceso, llamado el token primario del proceso. Este token es usado por las rutinas de validación de acceso de Windows NT cuando los hilos en el proceso hacen referencia a objetos protegidos.

#### ***MANEJADOR DE MEMORIA VIRTUAL:***

La arquitectura de memoria para Windows NT es un sistema de paginación virtual por demanda. El espacio de memoria virtual parecerá ser de 4 GB en tamaño, (2 GB reservados para almacenamiento de programas y 2 GB para almacenamiento de sistema). Windows NT puede ver hasta 4 GB de memoria física, si el hardware puede proveérselo.

El Manejador de memoria virtual mapea las direcciones virtuales en el espacio de direcciones del proceso a páginas físicas en la memoria de la computadora. Haciendo esto, oculta la organización física de la memoria a los hilos del proceso. Esto asegura que el hilo pueda acceder a la memoria de su proceso como lo necesite, pero no a la de otros procesos (porque cada hilo tiene un espacio de direcciones separadas, no puede ver ni modificar la memoria de otros procesos sin autorización).

#### ***FACILIDAD DE LLAMADAS A PROCEDIMIENTOS LOCALES (LPCF):***

Las aplicaciones y los ambientes de subsistemas tienen una relación cliente-servidor. Esto es, el cliente (aplicación) hace llamadas al servidor de ambientes (un subsistema) para satisfacer el requerimiento para algún tipo de servicio del sistema.

Para permitir esta relación. Windows NT, provee un mecanismo de comunicaciones entre el cliente y el servidor. El ejecutivo implementa una facilidad de message-passing llamada LCP. Trabaja en forma similar a RPC usando procedimientos de redes. pero LPC se optimiza al haber dos procesos corriendo en la misma computadora. El proceso de paso de mensaje queda oculto a las aplicaciones del cliente por funciones stubs provistas de bibliotecas especiales dinámicas (DLLs).

#### *MANEJADOR DE ENTRADA / SALIDA:*

El Manejador de E/S es la parte de Windows NT ejecutivo que maneja todas las entradas y todas las salidas del S.O.. Gran parte del rol de este Manejador es manejar las comunicaciones entre drivers. El Manejador de E/S soporta todos los sistemas de archivos de drivers, manejadores de dispositivos de hardware y manejadores de redes y provee un ambiente heterogéneo para ellos. Provee una interface formal que todos los manejadores o drivers pueden invocar. Esta interface uniforme le permite al Manejador de E/S comunicarse con todos los drivers de la misma manera sin tener conocimiento de cual de los dispositivos esta controlando actualmente. El Manejador de E/S también incluye rutinas especiales designadas para soporte de manejadores de sistemas de archivo, de dispositivos de hardware y de dispositivos de red.

El modelo de Windows NT para E/S usa una arquitectura de niveles que permite manejadores separados para implementar cada capa lógica distinta de un procesamiento de E/S.

*Manejador de cache:* la arquitectura de E/S incluye un único Manejador de cache que maneja la totalidad de las E/S del sistema por este método. El método cache es usado por un archivo de sistema para mejorar el rendimiento.

El Manejador de cache usa un archivo mapeado estrechamente integrado con el manejo de la memoria virtual de Windows NT. El Manejador de cache provee sus servicios a todos los archivos de sistema y componentes de red bajo el control del Manejador de E/S. El Manejador de cache puede hacer crecer dinámicamente y

disminuir el tamaño de la memoria cache a medida que varía la cantidad de memoria RAM disponible. Cuando un proceso abre un archivo que ya reside en la memoria cache, el Manejador de cache simplemente copia los datos de la cache al espacio de direcciones virtual del proceso y viceversa; la lectura y la grabación mejoran.

El Manejador de cache ofrece servicios como LAZY write y LAZY commit que pueden mejorar sobre el rendimiento de todos los archivos del sistema. LAZY WRITE es la habilidad de grabar los cambios en la estructura cache de los archivos que es más rápido que grabarlos sobre el disco y más tarde cuando hay una baja demanda de la unidad central del proceso, el Manejador cache envía los cambios al disco. LAZY COMMIT es similar al LAZY WRITE. En vez de marcar una transacción completada exitosamente en forma inmediata, la información declarada se coloca en la memoria cache y luego se graba al sistema de registros de archivos como un proceso back ground

*Sistemas de manejo de archivos:* en la arquitectura Windows NT de E/S, los manejadores de sistema de archivos son dirigidos por el Manejador de E/S. Windows NT soporta múltiples sistemas de archivos activos, incluyendo sistemas de archivos existentes como la F.A.T. Windows NT soporta sistemas de archivos F.A.T. y HPFS para resguardar la compatibilidad con los sistemas operativos MS-DOS, Windows 3.X y OS/2. La arquitectura de Windows NT de E/S no solamente soporta sistemas de archivos tradicionales sino que ha implementado su redirector de redes y servidor como un sistema de manejadores de archivos. Desde la perspectiva del Manejador de E/S no hay diferencia entre acceder archivos almacenados en una computadora remota vía red y acceder a aquellos almacenados localmente en un disco rígido. Además los redirectores y servidores pueden cargarse y descargarse dinámicamente como cualquier otro Manejador, y pueden coexistir múltiples redirectores y servidores a la misma computadora.

*Manejadores de dispositivos de hardware:* los manejadores de dispositivos de hardware también son componentes de la arquitectura de E/S. Todos ellos (como

manejadores de impresión, de Mouse, de disco) están escritos en el lenguaje de programación C, son direccionables a 32 bits y presentan seguridad al Multiprocesamientos.

Los manejadores de dispositivos acceden a los registros de hardware de los dispositivos periféricos a través de rutinas de soporte que provee el sistema operativo Windows NT. Hay un conjunto de estas rutinas para cada plataforma que soporta Windows NT. Como los nombres de las rutinas son iguales para todas las plataformas los manejadores de dispositivos para Windows NT son transportables entre cualquier tipo de procesadores

*Manejadores de red:* un tercer tipo de Manejador implementado como un componente de la arquitectura de E/S es el Manejador de red. Windows NT incluye capacidades de funcionamiento de red integradas y soporte para aplicaciones distribuidas.

**HAL** (Nivel de abstracción de hardware): virtual

Virtualiza las interfaces de hardware, haciéndolas transparentes al resto del S.O. Esto le permite portabilidad de una plataforma a otra.

**KERNEL:** Está en el núcleo de esta arquitectura de niveles y maneja las operaciones mas básicas de Windows NT. Es responsable del despacho de hilos (thread dispatching), organización del multiprocesamiento y manejo de excepciones de hardware.

**EJECUTIVO:** Colección de módulos del Kernel que provee servicios básicos del sistema operativo a los subsistemas del ambiente.

**SUBSISTEMAS DE ENTORNO:** Son servidores protegidos a modo usuario que ejecutan y soportan aplicaciones nativas en diferentes ambientes del SO.

**DESPACHADOR DE OBJETOS (Dispatcher Object):** Tiene una señal de estado (señalado ó no señalado) que controla el despacho y sincronización de las operaciones del sistema. Incluye: eventos mutantes, mutexes (acceso mutuamente excluido a un recurso), semáforos, hilos y relojes.

**OBJETOS DE CONTROL (Control Objects):** Se usan para controlar la operación del Kernel pero no afectan el despacho. Incluyen llamadas asíncronas a procedimientos, interrupciones, procesos y perfiles.

**HILO:** Unidad más pequeña de código ejecutable independiente.

**SERVICIOS DEL SISTEMA:** Interface entre los subsistemas del ambiente modo usuario y el modo Kernel.

### 2.3.12.8 Configuraciones de discos

Windows NT puede usar tres sistemas:

1. NTFS (NT File System)
2. FAT (File Allocation Table)
3. HPFS (High Performance File System OS/2)

Para una partición que no esta formateada el usuario puede elegir formatearla con NTFS o FAT. Si lo hace por medio del ultimo podrá acceder a la partición a través de DOS u OS/2; en el caso del primero sólo se podrá acceder a los archivos a través de Windows NT.

Además, si la partición ya existe en la maquina en la que deseamos instalar NT Server, se podrá convertir de FAT o HPFS a NTFS después de haber inicializado la instalación del servidor.

### 2.3.12.8.1 NTFS

#### VENTAJAS:

- Soporta completa seguridad Windows NT. El Superior puede decidir los tipos de acceso a los archivos y directorios.
- Guarda un LOG (archivo) de actividades para reconstruir el disco en el caso de problemas eléctricos.
- Soporta nombres de archivos y directorios de hasta 256 caracteres. Automáticamente genera archivos de corrección en DOS que pueden ser utilizados y compartidos.
- Permite un diseño de programa para correr bajo otro sistema operativo como el acceso MS-DOS a archivos de NTFS cuando esta corriendo Windows NT.

#### DESVENTAJAS:

- Reconocido tan solo por Windows NT. Cuando la computadora (Server de Windows NT) esta corriendo otro sistema operativo (DOS o OS/2) éste no puede acceder a la partición NTFS. Por ende no puede acceder a los archivos.
- Si el disco esta formateado para NTFS no podrá correrse DOS desde el disco rígido, pero si correrse aplicaciones de DOS usando Windows NT.

### 2.3.12.9 FAT

#### VENTAJAS:

- Permite el acceso a archivos y directorios por parte de otro Sistema Operativo. Es el mas comúnmente utilizado en las PC's.
- Permite compartir datos con otras particiones que contienen MS-DOS.

#### DESVENTAJAS:

- No provee seguridad Windows NT.
- Los archivos están restringidos a nombres de sólo 8 caracteres y 3 de extensión.
- No soporta archivos extremadamente grandes.

- Es menos "robusto" que NTFS ya que no permite la recuperación automática del disco.

#### 2.3.12.10 HPFS/HPFS386

##### VENTAJAS:

- Compatibilidad NT / OS/2.
- Soporta nombres largos de archivos.
- Provee mejor corrección de error que el sistema FAT.

##### DESVENTAJAS:

- No es muy adoptado.
- No provee la seguridad de archivo de NT.
- Las aplicaciones basadas en MS-DOS y Windows 3.1 no pueden acceder a archivos o directorios con nombres largos.

#### 2.3.12.11 Configuración de dominios, restricciones de usuarios y grupos

**DOMINIO:** Colección de computadoras que comparten la misma base de datos de usuarios, restricciones y niveles de seguridad.

Para controlar qué usuarios pueden acceder a la computadora y compartir archivos en un dominio NT el administrador de la red debe crear cuentas para los usuarios y especificar los grupos y niveles de seguridad.

El administrador, cuando configura el server NT, puede elegir su propio dominio o integrarse al dominio ya existente, siendo en el primer caso un Server Primario y en el segundo, Secundario. En este ultimo caso debe replicar la base de accesos desde el Server Primario de este Sistema Distribuido.

Antes de que una estación Windows NT o un Server puedan ser miembros del dominio y participar de su seguridad deben estar agregados al dominio; a las computadoras que se agregan se les da una cuenta en la base de seguridad; si la computadora que se agrega es el Server, este recibe una copia de la base de seguridad del dominio al cual se integro.

Existen tres formas de integrar una computadora a un dominio:

1. Una estación de trabajo puede estar integrada mientras se instala Windows NT y el Server puede ser integrado a un Dominio en la instalación del mismo pero solo puede ser realizado por el administrador.
2. Un administrador puede agregar una estación de trabajo al dominio.
3. Un administrador, o un par del mismo, puede agregar una cuenta para la computadora que queremos agregar; luego el usuario de la computadora debe tener, tanto la computadora como el dominio, dentro de la misma cuenta.

La sincronización de la base de datos de seguridad (dominio) la realiza y controla automáticamente el Server Primario, pero en el caso de que no se pueda sincronizar un determinado servidor ya que este se encuentra bajo, se debe realizar manualmente.

La Cuenta de cada Usuario consiste de la información que Windows NT usa para definirlo incluyendo su nombre, la descripción, una clave y los grupos a los cuales esta relacionado dicho usuario.

Un grupo es una cuenta que contiene otras cuentas, las cuales son conocidas como miembros; los Grupos proveen de una forma fácil de asignación de permisos a los usuarios sobre aplicaciones en común.

Un grupo puede ser global o local. En el primer caso, es un numero de cuentas de usuarios desde un dominio que están agrupados juntos bajo un mismo nombre de grupo. A un grupo global se le pueden asignar permisos y derechos con un dominio o una estación de trabajo. Un grupo local es un grupo que incluye usuarios y grupos

globales desde otros dominios, puede tener solamente permisos y derechos para su propia estación de trabajo.

### 2.3.12.12 Manejo de replicaciones de directorios

La replicación de directorios es la duplicación de un conjunto de directorios de un Server, llamado Server de exportaciones, a Servers o Estaciones de Trabajo llamados de importación, en el mismo u otros dominios. La replicación significa la tarea de mantenimiento de directorios y archivos idénticos en múltiples computadoras, porque solo con una copia de los datos, estos pueden ser mantenidos. Los archivos son replicados cuando se agregan al directorio de exportación y cada vez que se realiza un cambio se graba al archivo.

La replicación puede realizarse Exportando e Importando, debiendo especificar la ruta para poder realizarla.

La replicación ayuda al balanceo de cargas. Si hay un conjunto de archivos al cual muchos usuarios necesitan acceder, se puede repartir la carga en varias computadoras. De esta manera se configura un Server que actúa como exportador y utiliza un lugar para almacenar copias de los archivos y las otras computadoras actúan como importadoras.

Cada Server que actúa como exportador mantiene una lista de computadoras a las cuales serán exportados los archivos en sus respectivos subdirectorios y cada computadora importadora mantiene otra lista de computadoras conteniendo los subdirectorios que serán importados. Los servers exportadores pueden también exportar nombres de dominios y esto es conveniente para realizar la replicación de directorios para muchas computadoras, porque cada server exportador y computadora importadora necesitan especificar unos pocos nombres de dominio para exportar o importar, ya que de lo contrario tendrían una gran lista de computadoras. Esto último no siempre puede realizarse a través de una red WAN en la cual hay que especificar la

computadora de la cual se importaran los directorios. Cabe destacar que este tipo de servicio debe ser configurado ya que no viene por defecto.

### 2.3.12.13 Manejo de la seguridad

Utilizando NTFS además de las técnicas de Backup tradicionales, puede eliminarse toda posibilidad de pérdidas de datos.

Consiste de dos elementos básicos:

1. Prevención de la pérdida de datos por parte de accesos desautorizados. Windows NT incluye herramientas para limitar el acceso inicial al sistema, para archivos y directorios específicos y recursos (como impresoras). NT también incluye herramientas para monitoreo de los eventos de seguridad.
2. Prevención desde archivos dañados. Cuando se usa NTFS, Windows NT registra todas las transacciones de archivos, reemplaza a los sectores malos automáticamente y almacena una copia de toda la información vital. Además puede preservar la integridad del disco y asegurar una completa y rápida recuperación del sistema luego de haber sufrido una falla eléctrica u otro tipo de falla.

Windows NT también reserva una configuración previa del trabajo para asegurar que el usuario siempre arranque Windows NT; esto lo realiza para prevenir casos en los que se realizó algún cambio dentro de la configuración del mismo en una sesión de trabajo.

Windows NT permite además, crear un diskette de reparación de emergencia para el caso de que el sistema deba ser acondicionado por estar corrupto en el arranque.

El supervisor puede establecer una relación de confiabilidad entre dominios NT. En realidad son vínculos entre los dominios que permiten pasar mediante autenticación en la cual un usuario tiene una única cuenta en un dominio y aun puede acceder a toda la red integrada por varios dominios.

#### 2.3.12.14 Servicios de impresión

Windows NT permite compartir impresoras en su sistema; pueden estar en forma local o en red; cada impresora esta manejada por un programa llamado Print Manager, el que contiene su respectiva configuración y su estado.

El Print Manager le asigna un nombre a cada impresora, las computadoras que están corriendo Windows NT ven esos nombres cuando se conectan, mientras que las computadoras que corren MS-DOS requieren de un nombre que este dado por las convenciones (de no mas de 11 caracteres).

#### 2.3.12.15 Memoria virtual

Windows NT crea un archivo especial en el disco rígido, llamado archivo de paginación, que cumple con la función de hacer parecer que la computadora tiene una mayor Memoria Principal para hacer correr programas. Windows NT puede cambiar temporalmente los contenidos de memoria usados por un programa en el disco rígido para que otro programa pueda usar esa misma memoria. El tamaño de la Memoria Virtual esta relacionado con el de la Memoria Principal.

Cada aplicación, incluyendo DOS, Win32 y el subsistema de Windows, ejecuta en su propio espacio de memoria o maquina virtual. Si esta ejecutando una aplicación de DOS o Win32 en una maquina virtual, y otra aplicación se traba en otra maquina virtual, esto no interrumpirá a todo el sistema.

#### 2.3.12.16 Protocolos

##### 1. NETBEUI

El NetBios (Net BEUI) es un protocolo pequeño, eficiente y rápido para pequeñas redes LAN de alto desempeño. Se configura solo y se autocontrola. Se instala y relaciona automáticamente con el adaptador de red cuando se instala Windows NT.

## 2. DLC

El protocolo de Enlace de Control de Datos se usa principalmente para el acceso a las mainframe de IBM o impresoras conectadas directamente a la red. Este protocolo no se usa para trabajos en red con Windows NT.

Para conectarse y comunicarse con una computadora mainframe, este protocolo debe estar instalado y corriendo. Por ejemplo se usa una terminal emuladora como la 3270 en el Grupo de Accesorios para acceder a una computadora mainframe.

Para mandar a imprimir a impresoras conectadas directamente a la red, este protocolo debe estar instalado y corriendo sobre el servidor de impresión de la impresora. Las computadoras envían trabajos de impresión a la impresora usando DLC pero éstas no necesitan el protocolo, solo el servidor de impresiones lo requiere.

## 3. NWLink

El protocolo NetWare es una implementación de los protocolos IPX, SPX y NetBIOS que usan las redes de trabajo Novell NetWare.

NWLink posibilita la comunicación entre computadoras Windows NT o estaciones de trabajo NetWare. Dos interfaces de programación de aplicación de redes de trabajo (APIs) son soportadas para permitir esto: Windows Sockets y el NetBios.

La interface Windows Sockets es ideal para soportar la existencia de aplicaciones NetWare escritas para compilar con el IPX/SPX NetWare. La computadora Windows NT actúa como un Server de aplicación respondiendo a los paquetes de IPX o SPX enviados o recibidos de MS-DOS, Windows o estación de trabajo OS/2 NetWare.

Por ejemplo el Server de SQL corriendo en Windows NT tomo ventaja de la interface Windows Sockets a SPX para permitir a los clientes del Server SQL

---

NetWare comunicarse con el server que esta escuchando para requerimientos vía la interface Windows Sockets.

El soporte NetBIOS puede usarse para enviar y recibir paquetes de Novell NetBios entre una Estación de Trabajo NetWare corriendo bajo Novell NetBios y una computadora Windows NT o entre dos computadoras Windows NT. Se observará un aumento significativo del desempeño cuando todas las computadoras corren bajo Windows NT a causa del enlace de Microsoft para el protocolo de Novell NetBios.

El protocolo Microsoft incrementando Novell NetBios no solo conforma el standard simple de implementación de requerimiento y respuesta sino también toma automáticamente ventaja del conocimiento que implica la conversación con otra computadora que entienda los incrementos. El incremento implica eficiencia permitiendo al remitente enviar paquetes múltiples de NetBIOS antes de recibir un reconocimiento.

Los reconocimientos del NetBIOS son enviados por el Receptor después de recibirse el ultimo paquete.

La implementación sobre Windows NT separa los componentes del Novell NetBIOS desde los componentes del IPX/SPX. NWLink se refiere a IPX/SPX, y NWNBLINK se refiere a la implementación del Microsoft Incrementado de Novell NetBIOS.

Similar a la implementación de Windows NT TCP/IP NWLink se basa en un protocolo de STREAMS, el cual es un protocolo de interface y entorno originalmente desarrollados por UNIX que estandariza comunicaciones con estratos de software adyacente. Porque el entorno STREAMS soporta la interface conductora de trabajo en red (NDIS) debajo, NWLink puede coexistir con otros protocolos en la misma red.

### 2.3.12.5 Microsoft Windows 95.

Es un entorno multitarea dotado de una interfaz gráfica de usuario, que a diferencia de las versiones anteriores, Windows 95 no necesita del MS-DOS para ser ejecutado, ya que es un sistema operativo.

Este SO esta basado en menús desplegables, ventanas en pantalla y un dispositivo señalador llamado mouse. Una de las características principales de Windows 95 es que los nombres de los archivos no están restringidos a ocho caracteres y tres de la extensión, pueden tener hasta 256 caracteres para tener una descripción completa del contenido del archivo.

Además posee Plug and Play, una tecnología conjuntamente desarrollada por los fabricantes de PCs, con la cual un usuario puede fácilmente instalar o conectar dispositivos permitiendo al sistema automáticamente alojar los recursos del hardware sin la intervención de usuario.

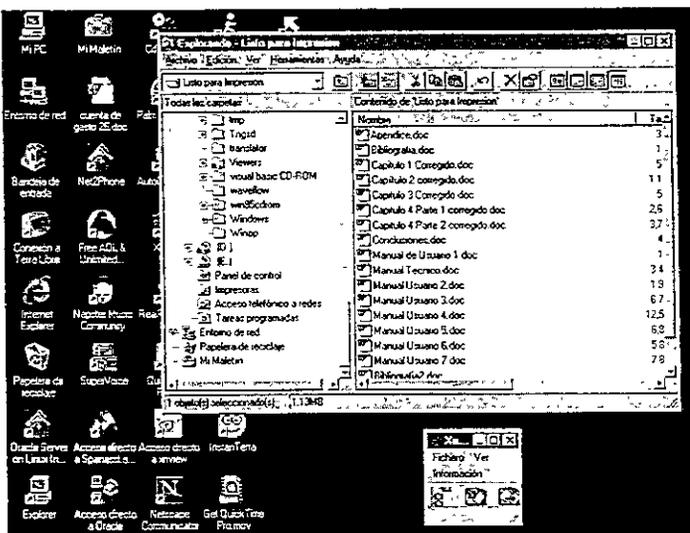


FIGURA 2.3.12.5.1 Aspecto del escritorio de Windows 95/98

### 2.3.12.6 LINUX

Linux es un sistema operativo semejante a UNIX, creado a principios de los noventas por Linux Torvalds con la asistencia de numerosos desarrolladores alrededor del mundo. Está públicamente disponible para bajarlo de Internet sin costo alguno, y el código fuente esta disponible para los desarrolladores.

El término Linux técnicamente se refiere sólo al kernel o corazón del sistema operativo, pero se hace referencia a él para describir al sistema operativo completo y una serie de aplicaciones que proporcionan mayor funcionalidad y facilidad de uso al usuario.

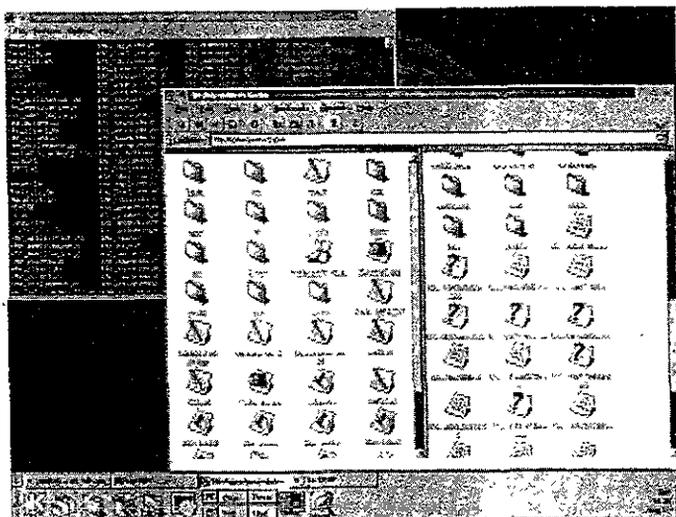


FIGURA 2.3.12.6.1 Aspecto del entorno gráfico KDE de Linux

Linux se apega a los estándares POSIX, que son una serie de estándares definidos en UNIX, por lo cual es compatible con los demás sistemas UNIX y similares.

En un principio, Linux fue desarrollado para las PC del hogar, pero en la actualidad es utilizado por una amplia variedad de plataformas, por ejemplo Macintosh, PowerPC, Amiga, Alpha, SPARC y muchas otras.

Cuando Linus Torvalds comenzó, a principios de los 90, a expandir las posibilidades del código MINIX, lo colocó para el dominio público bajo la Licencia Pública General de GNU (GPL General Public License). Esto motivó a que una serie de programadores alrededor del mundo contribuyeran con su desarrollo.

Debido a que este código está basado en estándares, su compatibilidad en especial con ambientes UNIX, así como Windows y Macintosh, es impresionante, razón por la que muchas empresas lo utilizan como servidor de archivos e impresoras para compartir información entre varias plataformas.

Como el código fuente está disponible para todos los programadores, también ha hecho posible que exista una gran comunidad dispuesta a desarrollar cuanta aplicación sea necesaria. Los errores que pudieran presentarse en el kernel se resuelven con rapidez, ya que al existir una gran base de programadores, las correcciones y nuevas implementaciones pueden ser probadas en una interminable cantidad de equipos, con diferentes configuraciones y necesidades; en consecuencia, las correcciones pueden ser más estables que las propuestas por otros sistemas operativos comerciales.

#### 2.3.12.7 Qué beneficios ofrece Linux.

Entre los beneficios podemos mencionar, la estabilidad, la capacidad de multitareas, el manejo de los recursos, en especial la memoria, la conectividad por medio de TCP/IP, las bibliotecas compartidas y la capacidad multiusuario. Por esto que gran cantidad de servidores de Internet se ejecutan en ambiente Linux, ya que con un solo CPU puede manejar una gran cantidad de usuarios y atenderlos con rapidez.

Por el alto nivel de procesamiento que se puede alcanzar con Linux y la amplia variedad de hardware en la que se puede instalar, este sistema operativo ha sido seleccionado por muchas empresas para cubrir sus necesidades específicas, ya que si requieren algo que no existe públicamente, pueden desarrollarlo e incluirlo en el núcleo del sistema.

Otra ventaja es que el sistema operativo y una gran cantidad de aplicaciones existentes para Linux son completamente gratuitos, lo que reduce de manera significativa los costos de propiedad de los equipos de cómputo en las empresas y los libra de las complejidades de licencias de uso de paquetes comerciales.

La compatibilidad de interconexión por red que se obtiene en Linux muchas veces supera y es más veloz y estable que la de algunos sistemas operativos comerciales, ya que logra comunicarse sin problemas con sistemas Macintosh, DOS, Windows 3.1x/95/98/NT/2000, Novell y OS/2 en una amplia variedad de protocolos: TCP/IP, IPX, ATM, módems, X.25, ISDN, Token-Ring, Appletalk y otros.

#### 2.3.12.8 Fragmentación de Archivos.

Es una condición por la que los archivos se dividen en el disco en pequeños segmentos separados físicamente entre sí. Esta condición es una consecuencia natural del crecimiento de los archivos y de su posterior almacenamiento en un disco lleno. Este disco ya no contendría bloques contiguos de espacio libre lo suficientemente grandes como para almacenar los archivos. La fragmentación de archivos no es un problema de integridad, aunque a veces puede ocurrir que los tiempos de acceso y de lectura aumenten si el disco está muy lleno y el almacenamiento se ha fragmentado incorrectamente. Existen productos de software para organizar u optimizar el almacenamiento de archivos.

En una base de datos, la fragmentación del archivo es una situación en la cual los registros no se graban en su secuencia de acceso óptima debido a las continuas

adiciones y eliminaciones de registros. La mayoría de los sistemas de bases de datos cuentan con utilidades que reordenan los registros para mejorar el rendimiento de acceso y recuperar el espacio libre ocupado por los registros borrados.

#### 2.3.12.9 Administración de Memoria.

Sea cual sea el esquema de organización del almacenamiento que se adopte para un sistema específico, es necesario decir que estrategias se deben utilizar para obtener un rendimiento óptimo. Las estrategias de administración del almacenamiento, determinan el comportamiento de una organización cuando se siguen diferentes políticas: ¿Cuándo se toma un nuevo programa para colocarlo en la memoria? ¿Se toma el programa cuando el sistema lo solicita específicamente o se intenta anticiparse a las peticiones del sistema? ¿En que lugar del almacenamiento principal se coloca el siguiente programa por ejecutar? ¿Se coloca los programas lo más cerca posible uno del otro en los espacios disponibles de la memoria principal para reducir al mínimo el desperdicio de espacio, o se colocan los programas lo más rápido posible para reducir al mínimo el tiempo de ejecución?.

### **Conclusión**

Luego de haber investigado y analizado se puede ver que se han desarrollado varios tipos de sistemas operativos con diferentes interfaces y categorías. Pero hemos podido observar que todos los sistemas operativos han sufrido cambios por parte de los programadores, y siguen evolucionando.

El diálogo entre el usuario y la máquina suele realizarse a través de una interfaz de línea de comandos o de una interfaz gráfica de usuario (GUI, siglas en inglés). Las interfaces de línea de comandos exigen que se introduzcan instrucciones breves mediante un teclado.

Las GUI emplean ventanas para organizar archivos y aplicaciones con iconos y menús que presentan listas de instrucciones. El usuario manipula directamente estos objetos visuales en el monitor señalándolos, seleccionándolos y arrastrándolos o moviéndolos con un mouse.

El uso de las GUI es más sencillo que el de las interfaces de línea de comandos. Sin embargo, la introducción de instrucciones con una GUI es más lenta, por lo que las GUI suelen tener la opción de emplear un sistema equivalente al de línea de instrucciones como alternativa rápida para los usuarios más expertos.

## 2.4 COMUNICACIONES

**Comunicación de Datos.** Es el proceso de comunicar información en forma binaria entre dos o más puntos. Requiere cuatro elementos básicos que son:

**Emisor:** Dispositivo que transmite los datos

**Mensaje:** lo conforman los datos a ser transmitidos

**Medio :** consiste en el recorrido de los datos desde el origen hasta su destino

**Receptor:** dispositivo de destino de los datos

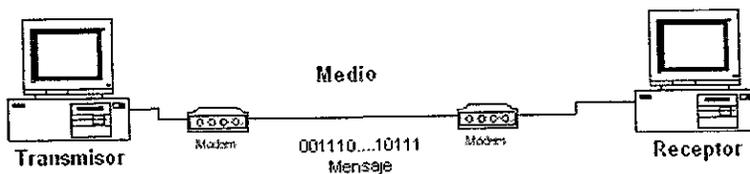


FIGURA 2.4.1 Receptor

**Bit:** es la unidad más pequeña de información y la unidad base en comunicaciones.

**Byte:** conjunto de bits continuos mínimos que hacen posible, un direccionamiento de información en un sistema computarizado. Está formado por 8 bits.

**Trama :** tira de bits con un formato predefinido usado en protocolos orientados a bit.

**Paquete :** fracciones de un mensaje de tamaño predefinido, donde cada fracción o paquete contiene información de procedencia y de destino, así como información requerida para el reensamblado del mensaje.

**Interfaces:** conexión que permite la comunicación entre dos o más dispositivos.

**Códigos:** acuerdo previo sobre un conjunto de significados que definen una serie de símbolos y caracteres. Toda combinación de bits representa un carácter dentro de la tabla de códigos. las tablas de códigos más reconocidas son las del código ASCII y la del código EBCDIC.

**Paridad:** técnica que consiste en la adición de un bit a un carácter o a un bloque de caracteres para forzar al conjunto de unos (1) a ser par o impar. Se utiliza para el chequeo de errores en la validación de los datos. El bit de paridad será cero (0=SPACE) o uno (1=MARK).

**Modulación:** proceso de manipular de manera controlada las propiedades de una señal portadora para que contenga la información que se va a transmitir.

**DTE (Data Terminal Equipment):** equipos que son la fuente y destino de los datos. comprenden equipos de computación (Host, Microcomputadoras y Terminales).

**DCE (Data Communications Equipment):** equipos de conversión entre el DTE y el canal de transmisión, es decir, los equipos a través de los cuales conectamos los DTE a las líneas de comunicación.

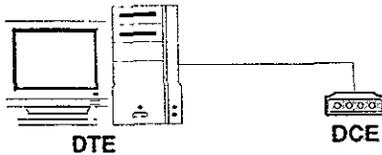


FIGURA 2.4.2 DCE y DTE

## 2.4.1 Medios, Formas Y Tipos De Transmisión

### Medios

**Aéreos:** basados en señales radio-eléctricas (utilizan la atmósfera como medio de transmisión), en señales de rayos láser o rayos infrarrojos.

**Sólidos:** principalmente el cobre en par trenzado o cable coaxial y la fibra óptica.

### Formas

**Transmisión en Serie:** los bits se transmiten de uno a uno sobre una línea única. Se utiliza para transmitir a larga distancia.

**Transmisión en Paralelo:** los bits se transmiten en grupo sobre varias líneas al mismo tiempo. Es utilizada dentro de la computadora.

La transmisión en paralelo es más rápida que la transmisión en serie pero en la medida que la distancia entre equipos se incrementa (no debe sobrepasarse la distancia de 100 pies), no sólo se encarecen los cables sino que además aumenta la complejidad de los transmisores y los receptores de la línea a causa de la dificultad de transmitir y recibir señales de pulsos a través de cables largos.

## Tipos

**Transmisión Simplex:** la transmisión de datos se produce en un solo sentido. siempre existen un nodo emisor y un nodo receptor que no cambian sus funciones.

**Transmisión Half-Duplex:** la transmisión de los datos se produce en ambos sentidos pero alternativamente, en un solo sentido a la vez. Si se está recibiendo datos no se puede transmitir.

**Transmisión Full-Duplex:** la transmisión de los datos se produce en ambos sentidos al mismo tiempo. un extremo que esta recibiendo datos puede, al mismo tiempo, estar transmitiendo otros datos.

**Transmisión Asíncrona:** cada byte de datos incluye señales de arranque y parada al principio y al final. La misión de estas señales consiste en:

- Avisar al receptor de que está llegando un dato.
- Darle suficiente tiempo al receptor de realizar funciones de sincronismo antes de que llegue el siguiente byte.

**Transmisión Síncrona:** se utilizan canales separados de reloj que administran la recepción y transmisión de los datos. Al inicio de cada transmisión se emplean unas señales preliminares llamadas:

- Bytes de sincronización en los protocolos orientados a byte.
- Flags en los protocolos orientados a bit.

Su misión principal es alertar al receptor de la llegada de los datos.

**Nota:** Las señales de reloj determinan la velocidad a la cual se transmite o recibe.

## 2.4.2 Protocolos

**Protocolo:** Conjunto de reglas que posibilitan la transferencia de datos entre dos o más computadoras.

**Arquitectura de Niveles:** el propósito de la arquitectura de niveles es reducir la complejidad de la comunicación de datos agrupando lógicamente ciertas funciones en áreas de responsabilidad (niveles).

### Características

- Cada nivel provee servicios al nivel superior y recibe servicios del nivel inferior.
- Un mensaje proveniente de un nivel superior contiene una cabecera con información a ser usada en el nodo receptor.
- El conjunto de servicios que provee un nivel es llamado **Entidad** y cada entidad consiste en un manejador (manager) y un elemento (worker).

## ESTÁNDARES

**OSI** (International Standards Organization)

**TCP/IP** (Transmission Control Protocol / Internet Protocol)

**IEEE** (Institute of Electrical and Electronic Engineers)

### OSI

En este modelo, el propósito de cada nivel es proveer servicios al nivel superior, liberándolo de los detalles de implementación de cada servicio. La información que se envía de una computadora a otra debe pasar del nivel superior al nivel inferior atravesando todos los demás niveles de forma **descendente**, dentro de la computadora que origina los datos.

A su paso por cada nivel a los datos se les adiciona información que será removida al llegar a su destino. La información adicionada se clasifica en:

**Información de Control**, dirigida a su nivel correspondiente en la computadora de destino. Cada nivel se comporta como si estuviera comunicándose con su contraparte en el otra computadora.

**Información de Interface**, dirigida al nivel adyacente con el cual se está interactuando. El objeto de esta información es definir los servicios provistos por el nivel inferior, y como deben ser accesados estos servicios. Esta información tras ser empleada por el nivel adyacente es removida.

**El modelo OSI se estructura en 7 niveles:**

1. **Nivel Físico:** este nivel dirige la transmisión de flujos de bits, sin estructura aparente, sobre un medio de conexión. Se encuentra relacionado con condiciones eléctricas-ópticas, mecánicas y funcionales del interfaz al medio de transmisión. A su vez esta encargado de aportar la señal empleada para la transmisión de los datos generados por los niveles superiores.

En este nivel se define la forma de conectarse el cable a las tarjetas de red, cuantos pines debe tener cada conector y el uso funcional de cada uno de ellos. Define también la técnica de transmisión a emplear para el envío de los datos sobre el medio empleado. Se encarga de activar, mantener y desactivar un circuito físico. Este nivel trata la codificación y sincronización de los bits y es el responsable de hacer llegar los bits desde una computadora a otra.

2. **Nivel de Enlace de Datos:** este nivel se encarga, en la computadora de origen, de alojar en una estructura lógica de agrupación de bits, llamada **Trama (Frame)**, los datos provenientes de los niveles superiores. **En la computadora de destino**, se encarga de agrupar los bits provenientes del nivel físico en tramas de datos

- (Frames) que serán entregadas al nivel de red. Este nivel es el responsable de garantizar la transferencia de tramas libres de errores de una computadora a otra a través del nivel físico.
3. **Nivel de Red:** es responsable del direccionamiento de mensajes y de la conversión de las direcciones lógicas y nombres, en direcciones físicas. Esta encargado también de determinar la ruta adecuada para el trayecto de los datos, basándose en condiciones de la red, prioridad del servicio, etc. El nivel de red agrupa pequeños fragmentos de mensajes para ser enviados juntos a través de la red.
  4. **Nivel de Transporte:** se encarga de la recuperación y detección de errores. Garantiza también, la entrega de los mensajes de la computadora originados en el nivel de aplicación. Es el nivel encargado de informar a los niveles superiores del estatus de la red.
  5. **Nivel de Sesión:** permite que dos aplicaciones residentes en computadoras diferentes establezcan, usen y terminen una conexión llamada **sesión**. Este nivel realiza reconocimientos de nombres y las funciones necesarias para que dos aplicaciones se comuniquen a través de la red, como en el caso de funciones de seguridad.
  6. **Nivel de Presentación:** determina el formato a usar para el intercambio de datos en la red. Puede ser llamado el traductor de la red. Este nivel también maneja la seguridad de emisión pues, provee a la red servicios como el de encriptación de datos.
  7. **Nivel de Aplicación:** sirve como ventana para los procesos que requieren acceder a los servicios de red.

## TCP/IP

Tiene como objetivos la conexión de redes múltiples y la capacidad de mantener conexiones aun cuando una parte de la subred esté perdida. A diferencia del modelo OSI, TCP/IP es software, lo que tiene como objetivo ser un modelo que pueda implementarse en cualquier tipo de red.

Los niveles físico y de enlace no son definidos en esta arquitectura, ya que pretende facilitar el intercambio de información independientemente de la tecnología y el tipo de subredes que atraviese, proporcionando un enlace transparente. Este protocolo sólo define tres capas que funcionarán en los niveles superiores a las capas física y de enlace, haciéndolo un modelo independiente del hardware en el que se implemente.

**Nivel de Internet.** Este nivel define el *Internet Protocol (IP)*, que provee el ruteo y control de congestión.

Los hosts pueden introducir paquetes en la red, los cuales viajan independientemente al destino. No hay garantías de entrega ni de orden.

**Nivel de transporte o de origen - destino.** Se encarga de controlar que los datos emanados de las aplicaciones lleguen correctamente y en orden a su destino, y permite que pares en los hosts de fuente y destino puedan conversar. Hay dos protocolos:

**Transmission Control Protocol (TCP).** Provee una conexión confiable que permite la entrega sin errores de un flujo de bytes desde una máquina a otra en *Internet*. Parte el flujo en mensajes discretos y lo monta de nuevo en el destino. Maneja el control de flujo.

**User Datagram Protocol (UDP).** Es un protocolo no confiable y sin conexión para la entrega de mensajes discretos. Se pueden construir otros protocolos de aplicación sobre UDP. También se usa UDP cuando la entrega rápida es más importante que la entrega garantizada.

---

**Nivel de aplicación.** Contiene la lógica necesaria para llevar a cabo las aplicaciones de usuario. Requiere de un módulo particular para cada tipo de aplicación.

### **Comparación de modelo OSI y TCP/IP**

- OSI define claramente las diferencias entre los servicios, las interfaces, y los protocolos.
  - Servicio: lo que un nivel hace
  - Interfaz: cómo se pueden acceder los servicios
  - Protocolo: la implementación de los serviciosTCP/IP no tiene esta clara separación.
- Porque OSI fue definido antes de implementar los protocolos, los diseñadores no tenían mucha experiencia con donde se debieran ubicar las funcionalidades, y algunas otras faltan. Por ejemplo, OSI originalmente no tiene ningún apoyo para broadcast.
- El modelo de TCP/IP fue definido después de los protocolos y se adecúan perfectamente. Pero no otras pilas de protocolos.
- OSI no tuvo éxito debido a
  - Mal momento de introducción: insuficiente tiempo entre las investigaciones y el desarrollo del mercado a gran escala para lograr la estandarización.
  - Mala tecnología: OSI es complejo, es dominado por una mentalidad de telecomunicaciones sin pensar en computadores, carece de servicios sin conexión, etc.
  - Malas implementaciones.
  - Malas políticas: investigadores y programadores contra los ministerios de telecomunicación.
- Sin embargo, OSI es un buen modelo (no los protocolos). TCP/IP es un buen conjunto de protocolos, pero el modelo no es general.

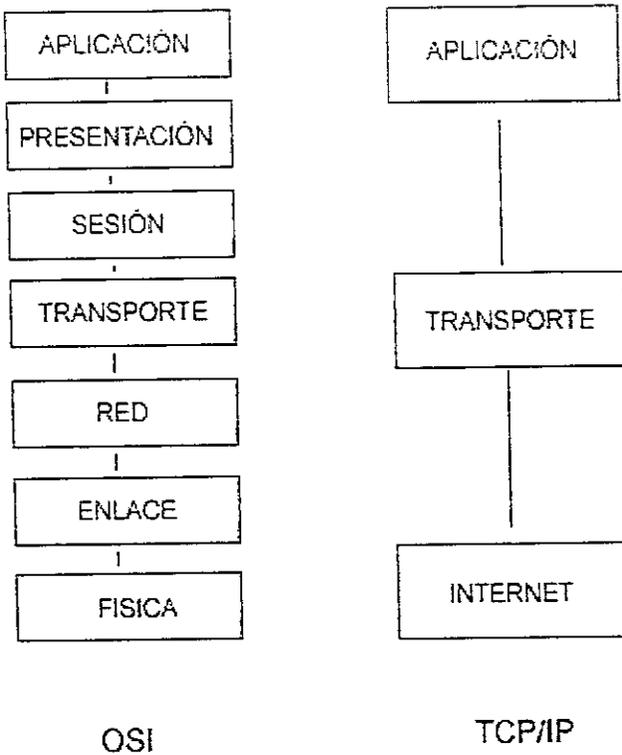


FIGURA 2.4.3 OSI y TCP/IP

**IEEE**

El modelo desarrollado por IEEE, también conocido como **el proyecto 802**, fue orientado a las redes locales. Este estándar está de acuerdo, en general con el modelo ISO, difieren principalmente en el nivel de enlace de datos. Para IEEE este nivel está dividido en dos subniveles:

**MAC (Medium Access Control):** subnivel inferior, provee el acceso compartido de las tarjetas de red al medio físico, es decir, define la forma en que se va a acceder al medio físico empleado en la red para el intercambio de datos.

**LLC (Logical Link Control):** subnivel superior, maneja la comunicación de enlace de datos y define el uso de puntos de interfaz lógico, llamado **SAP (Service Access Points)** de manera que otras computadoras puedan emplear el mismo formato para la comunicación con los niveles superiores independientemente del MAC empleado.

## **Clasificación de los Protocolos de Enlace de Datos**

### **De acuerdo a su estructura**

**Protocolos Orientados a Bit:** son aquellos protocolos en los cuales los bits por si solos pueden proveer información, son protocolos muy eficientes y trabajan en tramas de longitud variable.

**Protocolos Orientados a Byte:** son aquellos en los que la información viene provista por la conjunción de bytes de información y bytes de control.

### **De acuerdo a su disciplina de comportamiento**

**Protocolos de Sondeo Selección:** son aquellos que utilizan un DTE como nodo principal de canal. Este nodo primario controla todas las demás estaciones y determina si los dispositivos pueden comunicarse y, en caso afirmativo, cuando deben hacerlo.

**Protocolos Peer to Peer:** son aquellos en los cuales ningún nodo es el principal, y por lo general todos los nodos poseen la misma autoridad sobre el canal.

### **2.4.3 Redes WAN**

**Una Red Wan:** es una red de gran cobertura en la cual pueden transmitirse datos a larga distancia, interconectando facilidades de comunicación entre diferentes

localidades de un país. En estas redes por lo general se ven implicadas las compañías telefónicas.

### **Componentes Físicos**

**Línea de Comunicación:** medios físicos para conectar una posición con otra con el propósito de transmitir y recibir datos.

**Hilos de Transmisión:** en comunicaciones telefónicas se utiliza con frecuencia el termino "pares" para describir el circuito que compone un canal. Uno de los hilos del par sirve para transmitir o recibir los datos, y el otro es la línea de retomo eléctrico.

### **Clasificación Líneas de Comunicación**

**Líneas Conmutadas:** líneas que requieren de marcar un código para establecer comunicación con el otro extremo de la conexión.

**Líneas Dedicadas:** líneas de comunicación que mantienen una permanente conexión entre dos o más puntos. Estas pueden ser de dos o cuatro hilos.

**Líneas Punto a Punto:** enlazan dos DTE

**Líneas Multipunto:** enlazan tres o más DTE

**Líneas Digitales:** en este tipo de línea, los bits son transmitidos en forma de señales digitales. Cada bit se representa por una variación de voltaje y esta se realiza mediante codificación digital en la cual los códigos más empleados son:

**NRZ (Non Return to Zero) Unipolar:** La forma de onda binaria que utilizan normalmente las computadoras se llama **Unipolar**, es decir, que el voltaje que representa los bits varia entre 0 voltios y +5 voltios. Se denomina NRZ porque el voltaje

no vuelve a cero entre bits consecutivos de valor uno. Este tipo de código es inadecuado en largas distancias debido a la presencia de niveles residuales de corriente continua y a la posible ausencia de suficientes números de transiciones de señal para permitir una recuperación fiable de una señal de temporización.

**Código NRZ Polar:** este código desplaza el nivel de referencia de la señal al punto medio de la amplitud de la señal. De este modo se reduce a la mitad la potencia requerida para transmitir la señal en comparación con el Unipolar.

**Transmisión Bipolar o AMI (Alternate Marks Inverted):** es uno de los códigos más empleados en la transmisión digital a través de redes WAN. Este formato no tiene componente de corriente continua residual y su potencia a frecuencia cero es nula. Se verifican estos requisitos transmitiendo pulsos con un ciclo de trabajo del 50% e invirtiendo alternativamente la polaridad de los bits 1 que se transmiten. Dos valores positivos sin alternancia entre ellos serán interpretados como un error en la línea. los 0's son espacios sin presencia de voltaje. El formato Bipolar es en realidad una señal de tres estados (+V, 0, -V).

## INTERFACES

**RS-232 en 23 Y 9 Pines:** define una interfaz no balanceada empleando un intercambio en serie de datos binarios a velocidades de transmisión superiores a los 20,000 bps, opera con datos síncronos pero está limitada por una longitud de cable de aproximadamente 50 pies.

**V.35:** especifica una interfaz síncrono para operar a velocidades superiores a 1 Mbps. Este interfaz utiliza la mezcla de dos señales no balanceadas para control y de señales balanceadas para la sincronización y envío/recepción de los datos lo que facilita trabajar a altas velocidades.

## MÓDEMS

Un Módem es un dispositivo que convierte la señal digital en señal analógica y viceversa para posibilitar que el mensaje enviado por un DTE pueda llegar a otro(s) DTE's a través de líneas análogas.

Los Módems podemos seleccionarlos de acuerdo a:

1. La velocidad de transmisión
2. El tipo de línea que utiliza: dedicada, conmutada o ambas.
3. La modulación que emplea: FSK, PSK, DPSK, QAM, TCM.
4. Las posibilidades de compresión de datos para transmisión.
5. La modalidad de trabajo: punto a punto o Multipunto.
6. Si se instala interno o externo al equipo DTE.

En la practica el mercado de los módems crea dos grupos:

**Módems empleados en centros de transmisión** con una permanente o casi permanente actividad, las cuales cuentan con mecanismos sofisticados de diagnostico, control y administración centralizados y remotos.

**Módems de Escritorios** cuyo principal uso es la conexión a través de la red pública telefónica, con cierta regularidad pero nunca con carácter permanente ni con uso exhaustivo.

### Tipos De Modulación

**Modulación de Frecuencia (FSK, Frequency Shift Keying):** se utiliza en los modems de baja velocidad. Se emplea separando el ancho de banda total en dos bandas, los modems pueden transmitir y recibir datos por el mismo canal simultáneamente. El módem que llama se pone en el modo de llamada y el módem que responde pasa al modo de respuesta gracias a un conmutador que hay en cada módem.

**Modulación de Amplitud (ASK, Amplitud Shift Keying):** no se utiliza en solitario en comunicaciones de datos porque es muy sensible a interferencias de ruido eléctrico que pueden provocar errores en los datos recibidos.

**Modulación de Fase (PSK, Phase Shift Keying):** se codifican los valores binarios como cambios de fase de la señal portadora.

**Modulación Diferencial de Fase (DPSK, Differential Phase Shift Keying):** consiste en una variación de PSK donde se toma el ángulo de fase del intervalo anterior como referencia para medir la fase de cualquier intervalo de señal.

**Modulación de Amplitud de Cuadratura (QAM, Quadrature Amplitude Modulation):** se emplea en los módems más rápidos. Consiste en una combinación de PSK y ASK, es decir, se van a combinar las variaciones de amplitud en referencia al momento de fase en que ocurren con lo cual vamos a poder incluir más bits en los mismos hertz.

### **Compresión de Datos y Control de Errores**

**MNP (Microcom Network Protocol):** bajo estas siglas se agrupan un conjunto de protocolos que soportan interacción con aplicaciones de transferencia de datos. Esta dividido en las clases siguientes:

**Clase 2:** provee mecanismo de control de errores para transmisiones asincrónicas a 2400 bps con protocolos orientados a byte, la eficiencia anda por el 84%.

**Clase 3:** permite al módem aceptar datos en formato asincrónico y transmitirlos en modalidad sincrónica. La ventaja de este servicio es que limitan los bits de start y stop consiguiendo así un rendimiento de un 108%.

**Clase 4:** este servicio provee un ensamblamiento de paquetes adaptables. Posee un rendimiento de un 120%.

**Clase 5:** este servicio provee compresión de datos, negociación y duplexación, técnica que consiste en que los módems se conectan a la menor velocidad, para luego comenzar a negociar el uso de velocidades superiores.

### **Algoritmos de Compresión más usados**

**Codificación Huffman:** este algoritmo crea una tabla que codifica a los caracteres con longitud de bits variables, los más empleados en 4 bits y los menos empleados empiezan con 5 llegando hasta 11 bits.

**Codificación Run-Length:** se identifican secuencias repetitivas de al menos tres caracteres, enviándose al carácter seguido del número que indica la cantidad de veces que debe ser repetido ese carácter.

**V.42/V.42 BIS:** estos son los estándares de corrección de errores y compresión de datos respectivamente sugeridos por CCITT.

### **Supresión de Eco**

Posibilita la transmisión simultánea en ambos sentidos. Esta técnica solo es posible si el diseño del módem incorpora microprocesadores. La supresión del eco permite el uso de todo el ancho de banda de la línea para la transmisión simultánea en ambos sentidos del enlace.

### **Concentradores**

**Concentradores Análogos:** son dispositivos que permiten la comunicación entre un módem, conectado a un puerto de una computadora y varios módems conectados a DTE's en aplicaciones que usan protocolos de sondeo/selección. Con este tipo de concentrador, podemos bajar los costos de las líneas de comunicación. El concentrador análogo es el encargado de crear un equilibrio eléctrico entre los distintos enlaces.

**Concentradores Digitales:** también llamados Port-Sharing Devices, permiten que varios DTE's compartan un módem o un puerto de la computadora en aplicaciones que usan protocolos de sondeo/selección. Con este tipo de concentrador podemos ahorrar, dependiendo de como lo conectemos, puertos de un procesador de comunicaciones, host o módems requeridos para una conexión.

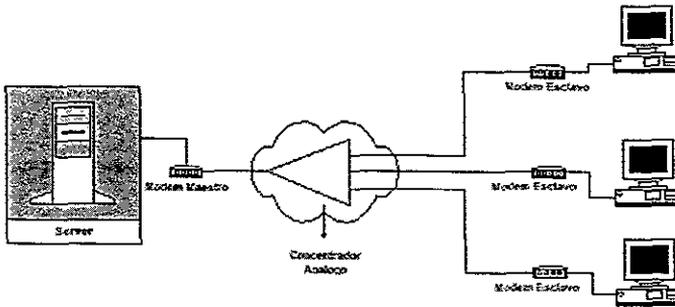


FIGURA 2.4.4 Concentradores

## Multiplexores

Dispositivos que permiten la combinación de varios canales de datos en un circuito físico.

**Multiplexor por División de Frecuencia:** divide el ancho de banda de una línea entre varios canales, donde cada canal ocupa una parte del ancho de banda de frecuencia total.

**Multiplexor por División de Tiempo:** aquí cada canal tiene asignado un periodo o ranura de tiempo en el canal principal y las distintas ranuras de tiempo están repartidas por igual en todos los canales. Tiene la desventaja de que en caso de que un canal no sea usado, esa ranura de tiempo no se aprovecha por los otros canales, enviándose en vez de datos bits de relleno.

**Multiplexor por División de Tiempo Estadísticos:** no le ofrece ranuras de tiempo a los canales inactivos y además podemos asignar prioridades a los canales.

## Procesadores de comunicación

Equipo cuya misión principal consiste en aliviar el trabajo de comunicaciones de la computadora central. Regula la comunicación tanto local como remota desde y hacia la *computadora central*.

Los Procesadores de Comunicación cargan, su propio sistema operativo desde una unidad de almacenamiento secundaria instalada en su interior o en una computadora central y es un nodo más en la red.

## TIPOS DE REDES WAN

**Conmutadas por Circuitos:** redes en las cuales, para establecer comunicación se debe efectuar una llamada y cuando se establece la conexión, los usuarios disponen de un enlace directo a través de los distintos segmentos de la red.

**Conmutadas por Mensaje:** en este tipo de redes el conmutador suele ser una computadora que se encarga de aceptar tráfico de las computadoras y terminales conectados a él. La computadora examina la dirección que aparece en la cabecera del mensaje hacia el DTE que debe recibirlo. Esta tecnología permite grabar la información para atenderla después. El usuario puede borrar, almacenar, redirigir o contestar el mensaje de forma automática.

**Conmutadas por Paquetes:** en este tipo de red los datos de los usuarios se descomponen en trozos más pequeños. Estos fragmentos o paquetes, están insertados dentro de informaciones del protocolo y recorren la red como entidades independientes.

**Redes Orientadas a Conexión:** en estas redes existe el concepto de multiplexión de canales y puertos conocido como **círculo o canal virtual**, debido a que el usuario aparenta disponer de un recurso dedicado, cuando en realidad lo comparte con otros pues lo que ocurre es que atienden a ráfagas de tráfico de distintos usuarios.

**Redes no orientadas a conexión:** llamadas Datagramas, pasan directamente del estado libre al modo de transferencia de datos. Estas redes no ofrecen confirmaciones, control de flujo ni recuperación de errores aplicables a toda la red, aunque estas funciones si existen para cada enlace particular. Un ejemplo de este tipo de red es INTERNET.

**Red Pública de Conmutación Telefónica (PSTN):** esta red fue diseñada originalmente para el uso de la voz y sistemas análogos. La conmutación consiste en el establecimiento de la conexión previo acuerdo de haber marcado un número que corresponde con la identificación numérica del punto de destino.

#### 2.4.4 Redes De Area Local

##### Componentes

**Tarjetas de Conexión a la red (NIC's):** tarjeta electrónica que conectan a las estaciones de trabajo a la red. Normalmente se insertan en una de las ranuras de expansión del motherboard de la microcomputadora suministrando de esta forma acceso directo a memoria (DMA). El NIC tiene las siguientes funciones:

- Forman los paquetes de datos
- Dan acceso al cable, con la conversión eléctrica y ajuste de velocidad
- Son el transmisor y el receptor de la estación
- Chequean las tramas para chequear errores
- Conversión Serie/Paralelo
- Identificación o dirección única en la red que permite saber cual es físicamente la terminal

**Estaciones de Trabajo:** PC's conectadas a la red a través de las cuales podemos acceder a los recursos compartidos en dicha red como discos, impresoras, módems,

etc. Pueden carecer de la mayoría de los periféricos pero siempre tendrán un NIC, un monitor, un teclado y un CPU.



FIGURA 2.4.5 Estación de trabajo

**Servidores:** Computadoras que proporcionan servicios a las estaciones de trabajo de la red tales como almacenamiento en discos, acceso a las impresoras, unidades para respaldo de archivos, acceso a otras redes o computadoras centrales.

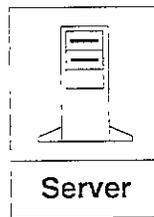


FIGURA 2.4.6 Server

**Repetidores:** dispositivos que generan la señal de un segmento de cable y pasan estas señales a otro segmento de cable sin variar el contenido de la señal. Son utilizados para incrementar la longitud entre conexiones en una LAN.

**Bridges:** consiste en un equipo que contiene dos puertos de comunicación, crea unas tablas en memoria que contienen todas las direcciones de MAC (direcciones de las tarjetas de comunicaciones), de ambos extremos, de tal manera que restringen el tráfico de datos de un segmento a otro, no permitiendo el paso de tramas que tengan como destino una dirección del mismo segmento al que pertenece la estación de origen. Es conveniente el uso de los mismos cuando requerimos la interconexión de dos LAN's locales o remotas.

**Routers:** son dispositivos que nos permiten unir varias redes( más de dos, a diferencia de los bridge), tomando como referencia la dirección de red de cada segmento. Al igual que los bridges, los Routers restringen el tráfico local de la red permitiendo el flujo de datos a través de ellos solamente cuando los datos son direccionados con esa intención.

**Brouters:** dispositivos con funciones combinadas de bridge y router. Cuando se configura se le indica la modalidad en la cual va a funcionar, como bridge o como router.

**Switch digital:** los ruteadores funcionan como dispositivos de acceso a una red conmutada, donde cada uno de los sitios se conectan al proveedor de servicio, usando la infraestructura de telecomunicaciones para transmitir otro tipo de información.

### Concentradores

**MAU ( Multistation Access Unit):** concentrador que permite insertar en el anillo o eliminar derivándolas, hasta 8 estaciones. El MAU detecta señales procedentes de las estaciones de trabajo, en caso de detectarse un dispositivo defectuoso o un cable deteriorado y elimina, derivándola, la estación en cuestión para evitar pérdidas de datos y del TOKEN.

**Hubs:** concentradores de cableado en estrella integrados por microprocesadores, memoria y protocolos como SNMP, características que lo convierten en un nodo inteligente en la red capaz de controlar y diagnosticar, incluso por monitoreo remoto.

**Switching Hub o Switch Ethernet:** divide la LAN en varios segmentos limitando el tráfico a uno o más segmentos en vez de permitir la difusión de los paquetes por todos los puertos. Dentro del Switch, un circuito de alta velocidad se encarga del filtrado y de permitir el tránsito entre segmentos de aquellos segmentos que tengan la intención de hacerlo.

## TOPOLOGÍA

**Estrella:** en este tipo de topología todas las estaciones de trabajo se conectan a una estación central que se encarga de establecer, mantener y romper la conexión entre las estaciones. En este tipo de red si cae la estación central cae toda la red.

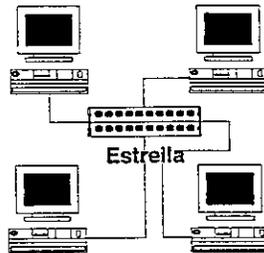


FIGURA 2.4.7 Topología de estrella

**Bus:** en esta topología todas las estaciones están conectadas al mismo cable. En una Red Bus, todas las estaciones escuchan todos los mensajes que se transfieren por el cable, capturando este mensaje solamente la estación a la cual va dirigido, que responde con un ACK o señal que significa haber recibido el mensaje correctamente.

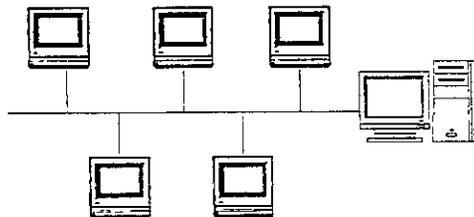


FIGURA 2.4.8 Topología de bus

**Anillo:** todos los nodos de la red están conectados a un bus cerrado, es decir, un círculo o lazo.

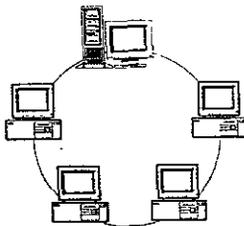


FIGURA 2.4.9 Topología de anillo

**Malla:** es usada en redes WAN, en donde los ruteadores se interconectan, y se utilizan para elegir la ruta óptima de la fuente al destino a través de la malla. Si algún enlace falla, no se interrumpe la conexión, porque se utiliza otra trayectoria para llegar al destino.

Un problema es que cada dispositivo requiere una tarjeta de interface para cada otro dispositivo en la red; aparte de que el monto total de cable es demasiado alto. Una red con topología puramente en malla no es considerada una buena solución práctica.

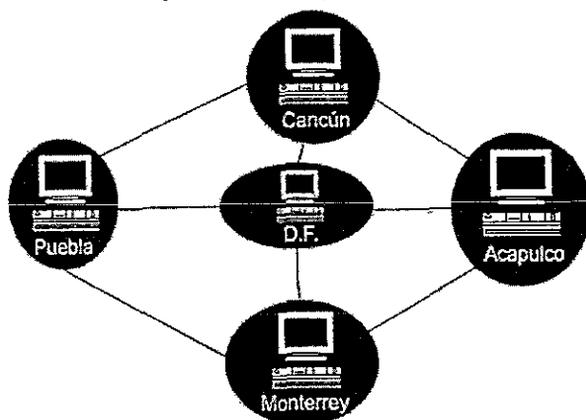


FIGURA 2.4.9 Topología de malla

## 2.5 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DE POWERBUILDER 5.0

La primera edición de PowerBuilder sólo funcionaba en una plataforma de 16 bits para Window 3.x, no ofrecía características de desarrollo orientado a objetos y solamente soportaba el estilo de **Single-Document Interface** (Interface de un solo documento SDI).

### 2.5.1 Versiones De Powerbuilder

En PowerBuilder 2, los programadores pudieron crear las aplicaciones de **Multiple-Document Interface MDI** (Interface de documentos múltiples) utilizando las ventajas de ventanas y menús.

PowerBuilder 3 introdujo objetos de usuario, barra de herramientas para MDI y muchas aplicaciones de Data Window, incluyendo informes de fichas cruzadas, etiquetas y otras opciones para realizar gráficos e ipresos comerciales.

PowerBuilder 4 consiguió mejoras el rendimiento, el conducto para datos, informes unitarios y combinados, el constructor de clases en C++, ampliaciones del soporte ole 2.0 y la barra de herramientas PowerTips (micro ayudas programada).

PowerBuilder 5 aportó Distributed PowerBuilder, código de máquina compilados ejecutables, barras de herramientas desplegadas y función de sobrecarga real. Aportó la sintaxis de anotación con punto para acceder a los datos y propiedades de DataWindow, nuevos controles de Windows 95 incluyendo Tab, TreeView, ListView, PictureLisBox y controles del RichTextEdit. El Internet Developer's Toolkit y las PowerBulder Foundation Class Libraires.

Con esta versión se puede compilar una aplicación a código de máquina para proporcionar un mejor desarrollo, nuevos estilos de Datawindow y la ayuda de los libros de PowerBuilder.

## 2.5.2 Qué Es Powerbuilder

PowerBuilder es una herramienta de desarrollo de aplicación orientada a objetos que permite construir aplicaciones de múltiples capas para correr en plataformas múltiples y actuar recíprocamente con varias Bases de Datos.

PowerBuilder es uno de un grupo de productos de Sybase que juntos proporcionan las herramientas para desarrollar aplicaciones cliente-servidor, distribuidas, y de Internet.

### Contenido de una aplicación en PowerBuilder

- Una interface de usuario. Menús, ventanas, y controles de la ventana que los usuarios manejan para dirigir la aplicación.
- Lógica de la aplicación. Código en eventos y funciones para codificar las reglas del negocio, como parte de la interface de usuario o en módulos separados llamar a objetos de usuario personalizados

### Las aplicaciones de PowerBuilder son manejadas por evento

En una aplicación de PowerBuilder, los usuarios controlan lo que pasa por las acciones que ellos toman. Por ejemplo, cuando un usuario pulsa un botón, escoge un artículo de un menú, o entra en datos en una caja de texto, se activan uno o más eventos. Por lo que se escribe código para especificar el proceso que debe pasar cuando se activan eventos.

Ventanas, controles, y otros componentes de la aplicación, contienen cada uno un juego de eventos predefinidos. Por ejemplo, cada botón tiene un evento clic asociado con él y cada caja de texto tiene un evento **Modified** (Modificado). La mayoría de los objetos tiene sus eventos predefinidos, pero también el programador puede definir sus propios eventos.

### 2.5.3 El Ambiente De Powerbuilder

El desarrollo en Powerbuilder permite crear objetos, ajustar sus propiedades y escribir scripts para definir procesos que se llevan a cabo cuando un evento ocurre.

Aplicaciones PowerBuilder:

- Proporciona un ambiente de usuario gráfico.
- Es orientado a objetos y maneja eventos.
- Puede acceder bases de datos localmente o en servidores de red

#### Ambiente de usuario gráfico

El Powerbuilder y las aplicaciones creadas con el programa, proporcionan al usuario una interface de usuario gráfica(GUI) con el sistema operativo. El Powerbuilder soporta los sistemas operativos Windows, Windows 95, Windows NT, Macintosh y UNIX .

#### Lenguaje PowerScript

Se escribe código propio de PowerBuilder usando PowerScript, el idioma de PowerBuilder. El código consiste en comandos de PowerScript, funciones, y declaraciones que desarrollan el proceso para responder a los eventos. Por ejemplo, el código para el evento clic del botón oprimido lee y muestra información de la Base de Datos y desarrolla un proceso en base a éstos. La ejecución de este evento puede causar que otro evento se dispare, por ejemplo que abra una ventana, etc.

PowerScript proporciona una gran variedad de funciones que se pueden usar en la aplicación. Hay una función por ejemplo, para abrir una ventana, una función para cerrar una ventana, una función para habilitar un botón, una función para actualizar la Base de Datos, y así sucesivamente. Además de poder construir funciones propias, el siguiente ejemplo muestra código en PowerScript.

```
gf_centra_ventana(this)
SetPointer(HourGlass!)
timer(0.2)
sle_usuario.text = 'E'
sle_password.text = 'QE'
sle_usuario_informix.text = 'dba'
sle_password_informix.text = 'sql'
```

### Proceso de manejo de eventos:

En un ambiente de este tipo el usuario selecciona objetos que están dentro de ventanas para controlar la secuencia de procesamiento o flujo de programa. La aplicación no realiza ningún proceso hasta que el usuario inicia la acción, por ejemplo: al oprimir la tecla de "cerrar" en una ventana, es un evento que indica a la aplicación que debe ejecutar la instrucción **close** (cierra) la ventana. En la figura 2.5.1 se muestra algunos eventos del objeto window.

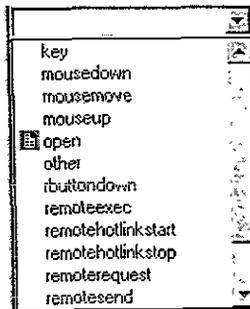


FIGURA 2.5.1 Eventos de un objeto

### Conceptos de programación orientada a objetos:

El desarrollo orientado a objetos ve el análisis, diseño e implementación de sistemas desde la perspectiva de objetos, los cuales tienen propiedades y comportamientos

específicos llamados métodos, asociados con ellos. Este concepto también toma en cuenta las técnicas de programación orientada a objetos como son encapsulación, herencia, y polimorfismo, estas técnicas permiten hacer una aplicación extensible, reusable y mejor.

**¿Qué es un objeto?:** Un objeto representa una sola cosa, los objetos son componentes discretos que combinan características (datos) y comportamientos (métodos).

**Los objetos tienen:**

- Un nombre, para referirnos a este.
- Propiedades, para describir el objeto.
- Métodos para codificar su comportamiento.

**Propiedades de los objetos:** Propiedades son las características de un objeto, es decir, determina como éste se ve, sin embargo, algunas propiedades definen el comportamiento de un objeto.

**Métodos:** Pueden ser eventos y funciones.

**Eventos.** Son notificaciones de las acciones que se llevan a cabo. Aplicaciones, controles, menús, windows y objetos de usuarios, así como objetos no visuales, cada uno viene con un conjunto de eventos predefinidos. Los eventos están ubicados en el sistema operativo o en otros eventos de Powerbuilder.

Ejemplos de eventos que el PowerBuilder incluye: Clicked, Constructor, Open, RButtonDown, RetrieveStart. En éstos se codifica para que al abrir una ventana realice algún proceso.

**Funciones:** El PowerBuilder incluye un conjunto de funciones de sistema, como el Open() y Close( ), que están disponibles para todos los objetos de PowerBuilder.

Cada objeto de PowerBuilder tiene su propio conjunto de funciones predefinidas. Por ejemplo, el objeto ventana cuenta con las funciones Show y Hide, mientras que el control DataWindow tiene las funciones Retrieve y Update.

Ejemplo: el evento clicked para el "commandbutton" ejecuta un script que cierra la ventana.

#### **2.5.4 Arquitectura Cliente / Servidor.**

Las aplicaciones del PowerBuilder se encuentran dentro de un ambiente cliente / servidor. Arquitectura cliente / servidor divide los procesos entre los clientes y los servidores de una red.

Un cliente presenta la interface de usuario, genera requerimientos para datos y puede procesarlos. El servidor responde a los requerimientos del cliente y asegura la integridad de los datos.

La arquitectura cliente / servidor tiene tres principales componentes:

- a) La aplicación Cliente corre en una o más estaciones de trabajo ( usualmente PCs, computadoras Macintosh o estaciones de trabajo UNIX ).
- b) Bases de datos y servidores de aplicaciones.
- c) Redes de transferencia de datos entre clientes y servidores.

Ejemplo: Una aplicación usando la arquitectura cliente / servidor requiere de servicios desde un servidor de base de datos relacional, a través de declaraciones SQL (Lenguaje de Consulta Estructurado).

El cliente envía un mensaje con los requerimientos de datos a través de la red del servidor. El servidor recibe los requerimientos, trae los datos y pasa la información de regreso al cliente de nuevo por la red.

Una vez que el cliente recibe los datos, puede ocurrir un proceso adicional en la máquina del cliente.

### 2.5.5 Aplicaciones Desarrolladas Con Powerbuilder

Estas van desde aplicaciones propias con PowerBuilder, Internet, Distribuidas y Cruzadas como se menciona a continuación:

#### Aplicaciones de Internet

Se pueden desarrollar aplicaciones de PowerBuilder que corren en el Web. Estas aplicaciones toman ventaja de la siguiente tecnología:

- Web PB. Incluye funciones de objeto que pueden ser llamados por **browsers** (navegadores).
- DataWindow plugin. Permite a un navegador mostrar reportes de PowerSoft.
- *PowerBuilder window plugin and PowerBuilder window ActiveX*. Permite al navegador mostrar ventanas de PowerBuilder.
- Componentes DataWindow ActiveX y DataWindow JavaBeans. Usa objetos DataWindow creados en PowerSite or PowerJ.
- Clases JavaBeans proxy. El cliente Java accesa el servidor de PowerBuilder.

#### Aplicaciones Distribuidas

- Centralizar la lógica del negocio en los servidores (Jaguar CTS, MTS, y los servidores de PowerBuilder distribuidos)

- Particionar las funciones de la aplicación entre el cliente y el servidor reduciendo la carga de trabajo en el cliente
- Aplicaciones escalables que son fáciles mantener.

### **Desarrollo en plataformas cruzadas**

Se puede desarrollar una aplicación que usa PowerBuilder en Windows y se puede ver la misma aplicación en UNIX, desarrollando la misma aplicación al mismo tiempo. Se puede compartir objetos usados en la aplicación aunque sean diferentes plataformas.

### **Conectividad de Base de Datos**

PowerBuilder proporciona acceso fácil a información corporativa guardada en una variedad de Base de Datos.

Pueden accederse datos a través de interfaces de PowerBuilder como son ODBC o JDBC, a través de un nativo o a conexión directa a la Base de Datos.

### **2.5.6 Ayuda De Línea Y Documentación**

Se puede seleccionar la ayuda por medio del botón F1 o el menú, además de instalar los manuales completos para consultarlos.

### **2.5.7 Ambiente De Diseño Y Desarrollo De Power Builder 5.0**

El ambiente de desarrollo para Power Builder 5.0, esta conformado por una serie de barras de herramientas, compuesta básicamente por una barra principal, una secundaria y dependiendo del objeto a editar se tiene una tercera barra de herramienta. Como la mostrada en la figura 2.5.2

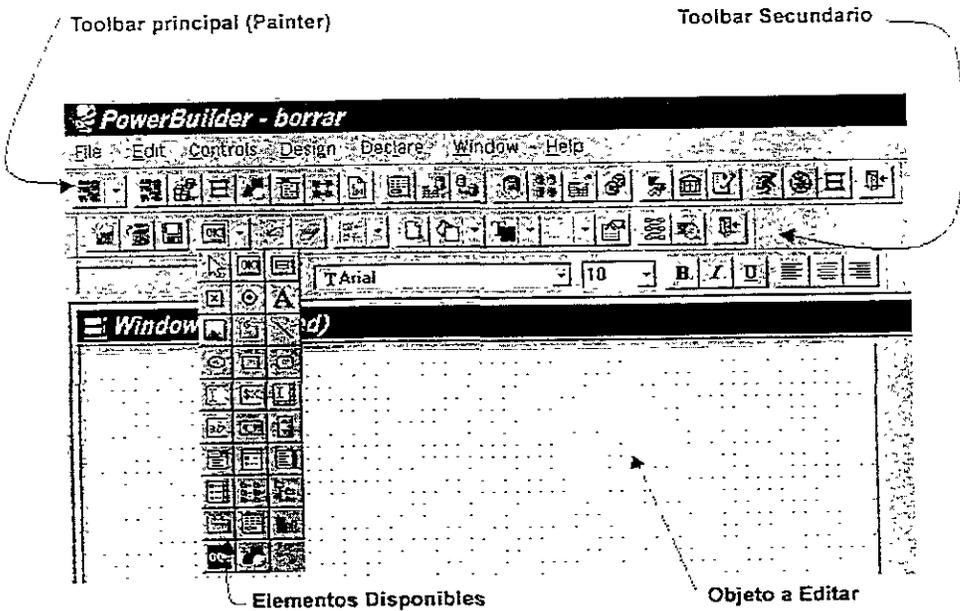


FIGURA 2.5.2 Ambiente de Desarrollo

Como se muestra en la figura 2.5.2 dependiendo del objeto a diseñar tendremos los elementos que se podrán utilizar en el objeto, así como su ambiente de programación.

### Objetos de PowerBuilder

En el PowerBuilder puede crear y modificar objetos usando herramientas gráficas llamadas "painters", por ejemplo, puede crear y modificar las definiciones de los objetos empleando las Aplicaciones painter, objetos window usando el Window painter, objetos de Base de datos Window utilizando el DataWindow painter, etc. Para acceder al "painter". basta con oprimir el icono que se encuentra en la barra principal de herramientas conocida también como Powerbar.

## Componentes del painter

Un "painter" incluye una o más barras de herramientas, conocidas como PaintBars, un menú, Micro-ayuda, y un espacio de trabajo. En el PaintBars se tiene el PowerBar que es la barra de herramientas que contiene los objetos principales de PowerBuilder como se muestra en la figura 2.5.3



FIGURA 2.5.3 Painter

## PowerBuilder Painters

### *Painters de los Objetos de Librería.*

Application Painter.	Crea y mantiene la definición de la aplicación.
DataWindow Painter.	Crea y mantiene objetos DataWindow
Function Painter.	Crea y mantiene funciones definidas por el usuario
Menu Painter.	Crea y mantiene menús.
Query Painter.	Crea y mantiene sentencias SQL usadas en la creación de objetos DataWindow
Structure Painter.	Crea y mantiene estructuras
User Object Painter.	Crea y mantiene objetos definidos por el usuario
Window Painter.	Crea y mantiene controles y ventanas

### *Painters de Soporte*

Database Painter.	Crea y mantiene tablas, índices, vistas e información de seguridad
Library Painter.	Crea y mantiene librerías y entradas de librería
Preference Painter.	Crea y mantiene variables de ambiente de PowerBuilder
Power Script Painter	

---

Power Script Painter. Crea y mantiene scripts de objetos y controles en una aplicación

#### *Painters de Base de Datos Adicionales*

Select Painter. Pinta la sentencia SELECT para recuperar de una Tabla de Base de Datos

SQL Painter. Pinta una sentencia SQL

View Painter. Crea y mantiene vistas de columnas en una tabla de base de datos y otras vistas

NOTA: Si se planea exportar objetos se debe asegurar que los primeros ocho caracteres del nombre de cada uno de los objetos específicamente identifican el objeto. La operación de exportación coloca cada objeto en un archivo separado y usa los primeros ocho caracteres del nombre del objeto como nombre por default del archivo.

Application Painter. Punto de entrada para una colección de ventanas organizadas y otros objetos que realizan actividades. También hay objetos no visuales que mantienen las características por omisión de una aplicación.

Path de Búsqueda de Librerías. Los objetos que componen una aplicación de PowerBuilder pueden ser guardados en una librería PB (PBL). Alternativamente puede haber objetos que quieran ser guardados en diferentes PBL's. A través del Path de búsqueda de librerías uno o más PBL's pueden ser ligadas y así usar objetos, ventanas, funciones, etc. de otras librerías sin tener que duplicar el código.

Ejemplo:

c : \cursopb\curso.pbl

e : \examenpb\examcn~pbl

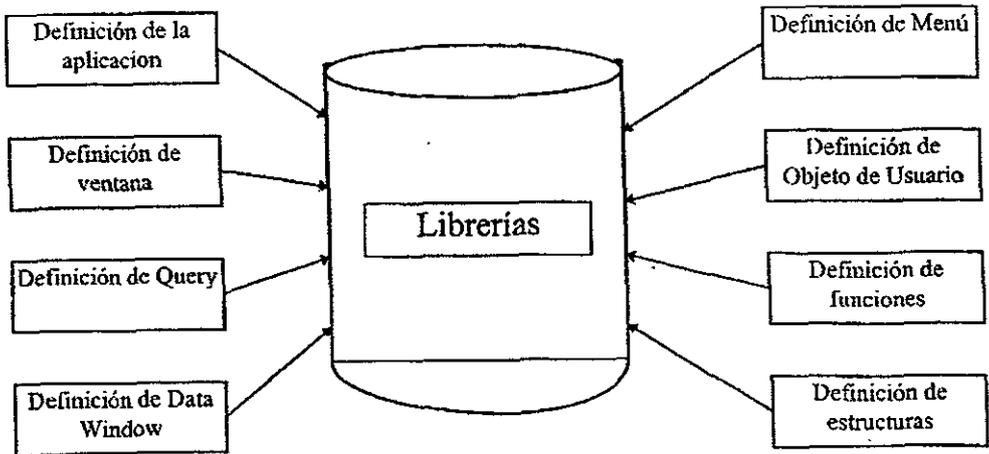


FIGURA 2.5.4 Relación de Librerías

Creación de una aplicación

Pasos para construir una aplicación

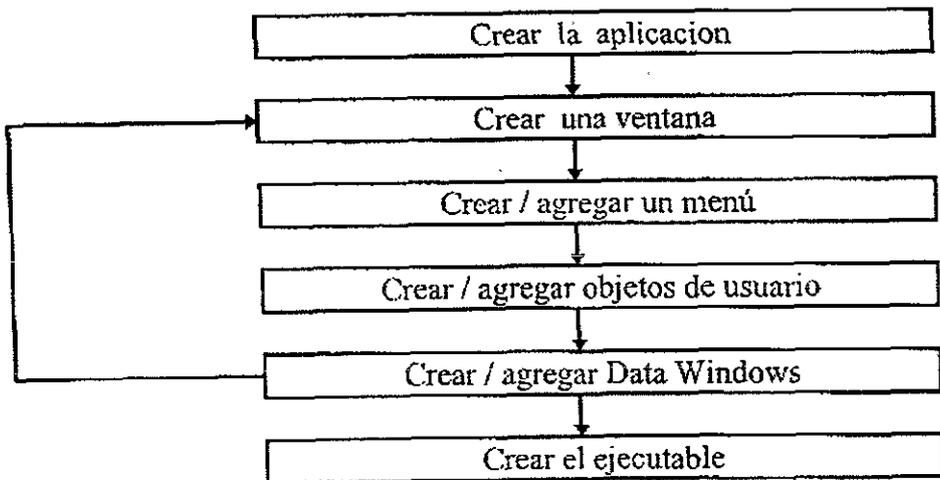


FIGURA 2.5.5 Pasos para crear una aplicación

## **Objetos Principales de PowerBuilder**

Objeto de aplicación: es un punto de entrada a la aplicación, éste es guardado en una librería PBL, así mismo se almacenan otros objetos.

¿Qué hace el objeto de aplicación? Define el nivel de función de la aplicación, contiene las librerías en donde se encuentran los objetos, el tipo de letra y que procesos pueden ocurrir cuando comienza y termina la aplicación, si existe un error durante la ejecución se dispara un evento que para la ejecución.

## **Aplicación Windows**

Es un programa escrito para el ambiente windows. Estas aplicaciones usan menús pull-down, iconos y procedimientos de point-and-click.

## **Manejador de Eventos**

En una aplicación PowerBuilder el usuario y no el programador, controla la secuencia de los procesos. En otras palabras el proceso depende de la acción que tome el usuario y no de un orden preestablecido. Por esta razón se dice que se manejan eventos.

## **Objetos**

Componentes que combinan características y acciones. Las características son atributos y las acciones son eventos.

Los objetos se comunican entre sí a través de mensajes.

El objeto que envía el mensaje, no sabe si lo recibió otro objeto; el objeto receptor recibe el mensaje y sabe lo que tiene que hacer por sus acciones inherentes.

## Atributos

Son las características de los objetos y determinan como van a verse.

## Evento

Cuando un usuario selecciona un objeto, introduce o modifica información en un objeto, un evento ocurre. Un evento ocurre cuando el usuario realiza una acción con el Mouse o teclado. También puede ser causado por sentencias de PowerScript y por el sistema mismo.

Eventos más comunes:

- Clicked
- DoubleClicked
- Modified
- Open
- Close
- DragAndDrop

Los eventos provocan que un script sea ejecutado. Por ejemplo, un evento clicked ocurre cuando el usuario hace click con el Mouse y el apuntador está en un CommandButton. Se puede codificar un script para el CommandButton para responder a este evento clicked.

## Script

Un script define un proceso a ejecutar. Cuando un evento ocurre PowerBuilder ejecuta el script para ese evento. Un script es escrito en el lenguaje PowerScript.

## 2.5.8 Definición De Powerscript Lenguaje

Es un lenguaje de Alto Nivel basado en objetos, el cual se usa para construir scripts.

- Power Scripts tiene más de 300 funciones predefinidas (tales como OPEN, HIDE, MAX).
- Comandos de Procedimiento (tales como IF.. .THEN)
- Sentencia SQL de manipulación de datos (tales como SELECT, INSERT, UPDATE)

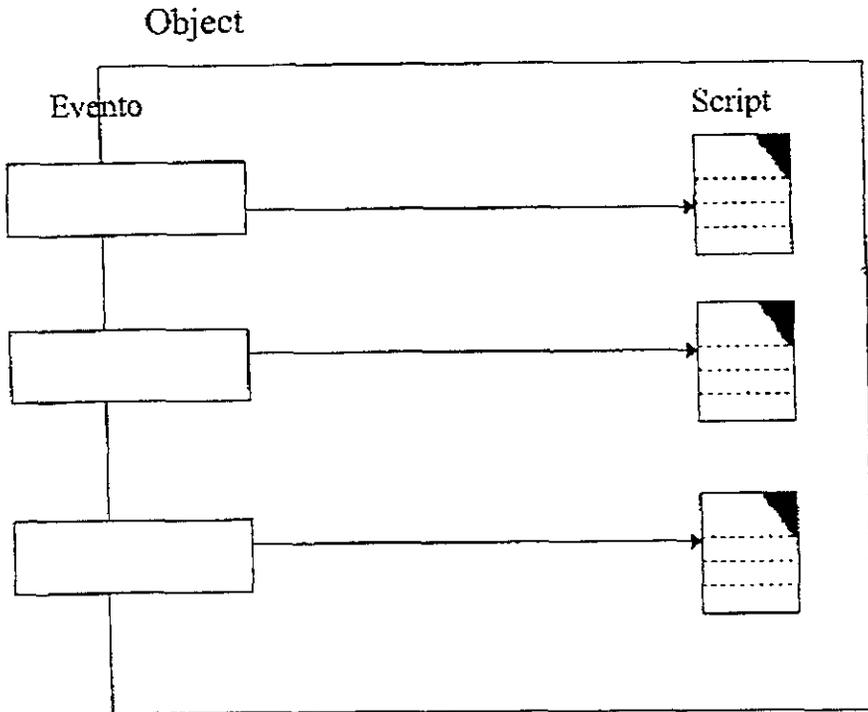


FIGURA 2.5.6 Relación de Evento-Script

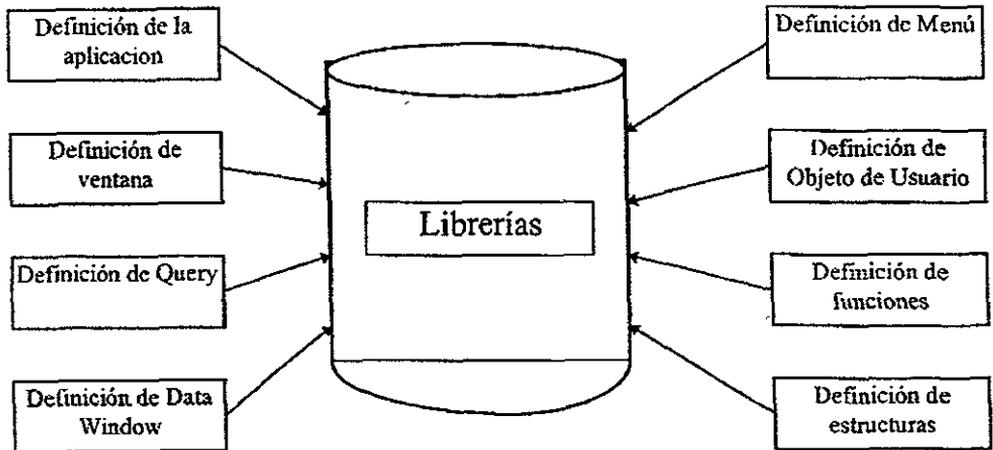


FIGURA 2.5.7 Conjunto de Librerías

### Objeto Ventana

Es un bloque principal de construcción de una aplicación y la interfaz principal que establece la comunicación del usuario con PB. Una ventana puede desplegar información, pedírsela al usuario y responder a acciones del Mouse y del teclado.

Una ventana consiste de:

- Propiedades. Definen la apariencia y funciones de la ventana, por ejemplo la ventana puede tener un título o puede ser minimizada, etc.
- Eventos. Son disparados por acciones del usuario
- Controles. Son colocados en la ventana.

Las ventanas pueden desplegar información, pedir información al usuario y responder a acciones del Mouse y del teclado.

### Tipos de ventanas.

Cuando se crea una ventana se define su apariencia general y el comportamiento:

- Estilo, tipo, tamaño, posición, color, etc.
- Controles, donde el usuario interactúa con la aplicación por medio de los controles encontrados en la ventana, como pueden ser radiobutton, single line edit, dropdown list box, commandbutton, etc. Como se muestra a continuación.

### Estilos de Ventanas.

Cada Ventana tiene un estilo que determina como se verá y como se comportará. Atributos de la Ventana. Cada Ventana tiene atributos que usted puede usar para especificar la apariencia y comportamiento de una ventana.

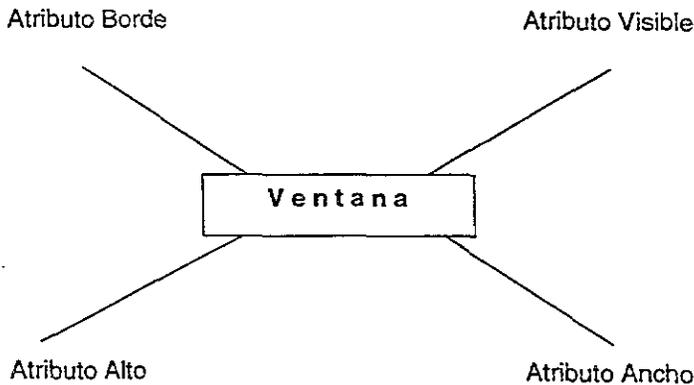


FIGURA 2.5.8 Atributos

### Componentes de una Ventana.

Se puede asociar un menú con una ventana y se pueden colocar objetos PB (excepto el objeto application) y controles en una ventana. Cualquier cosa colocada en una ventana se conoce como control. Uno determina el estilo, tamaño y posición de cualquier control en la ventana.

## Tipos de Ventana

### Principal (Main)

Uso. Ventana inicial de una aplicación.

Característica. Opera independientemente a otras ventanas

Puede tener una barra de título

### Hija (Child)

Uso. Es dependiente de la ventana padre. Puede ser usada cuando se desea que la ventana padre despliegue datos y la ventana hija despliegue una gráfica.

Características. Son subordinadas a la ventana padre.

Pueden existir únicamente con ventanas padres y se cierran al mismo tiempo que las ventanas padres.

Se limita a la circunscripción de la ventana padre.

La posición de la ventana es relativa a la ventana padre.

Aparecen como iconos con ventanas padres cuando se minimizan

Puede tener una barra de título

### Movable Popup

Uso. En las ayudas en línea

Características. Dependen de ventanas padres

Pueden desplegarse fuera de ventanas padres

Cuando se minimizan aparecen como iconos fuera de las ventanas padres.

Nunca esta detrás de la ventana padre

Puede tener una barra de título o menú

### De respuesta (Response)

Uso. Obtiene y/o brinda información al usuario

Requiere un mínimo de respuesta del usuario para trabajar.

Características. Ventanas que no pueden ser minimizadas

El usuario no puede usar otras ventanas en la misma aplicación

El usuario puede ir a otra aplicación en windows.  
Se mantiene activa hasta que el usuario responda.

FIGURA 2.5.9 Ejemplo de Ventana

## Objeto DataWindow

Combinan un acceso de datos inteligente con una *interface al usuario*, puede ser usada para desplegar o capturar datos. La DataWindow es un objeto único de PowerBuilder

Entre los objetos más importantes la Datawindow con diferentes estilos de presentación, como se muestra en la figura 2.5.10

El data **Source** (datos fuente) nos permite seleccionar la fuente de datos, mientras con la presentación del wizard se selecciona el estilo de edición de los datos.

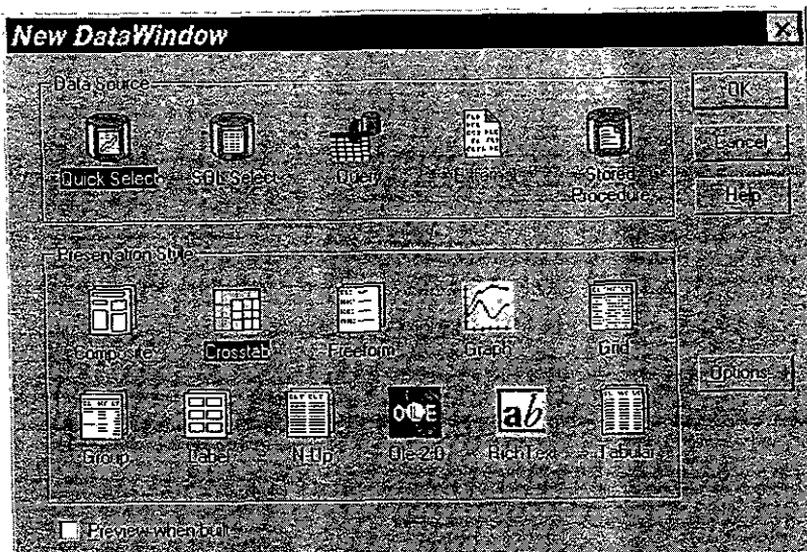


FIGURA 2.5.10 Objeto wizard para la presentación de los datos

### Objeto menú

Objeto visual que permite al usuario acceder una lista de comandos, opciones o diferentes caminos para realizar una acción. Son una lista de **items**(opciones) que el usuario puede seleccionar desde una barra de opciones para la ventana activa. Proporcionan a través de un comando una manera de desarrollar una tarea, por ejemplo, el menú Edición o el abrir una ventana. La selección de un menú item puede realizarse por medio del ratón, teclado o teclas aceleradoras. A continuación se muestra un diseño de menú.

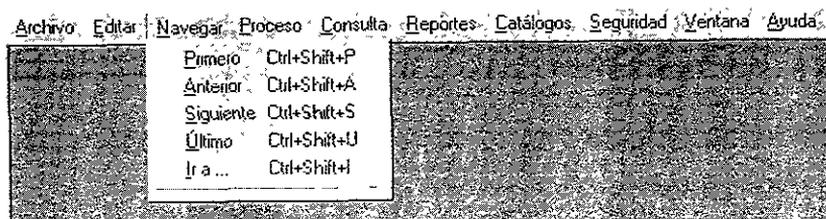


FIGURA 2.5.10 Ejemplo de menú

## **Objeto Query**

Es un objeto no visible que contiene comandos de SQL y que puede ser asociado a un objeto DataWindow.

## **Objeto de Usuario**

Son objetos que pueden ser construidos por el programador para realizar procesos frecuentes en su aplicación.

## **Funciones**

Consiste en un grupo de enunciados PB para hacer un proceso y que este regrese un valor. Existen funciones a nivel objeto y que se encuentran encapsuladas en él.

## **Estructuras**

Colección de una o varias variables que están agrupadas bajo un mismo nombre permite hacer referencia como una sola unidad.

## **Documento de interfaces múltiple (MDI frame)**

Uso. Para manejar más de un documento a la vez, presentando un amplio rango para el manejo de datos.

Característica.

Presenta un amplio rango para el manejo de datos

Presenta una interfaz integrada al usuario

Realiza la mayor parte de sus actividades en sus menús.

Debe estar asociada a un menú

Permite tener abiertas múltiples ventanas

El usuario puede activar cualquier ventana

Puede contener múltiples instancias de la misma o de diferente tipo de ventana dependiendo de la ventana que se active

Las ventanas pueden presentarse en cascada, apiladas o a lo largo de la ventana completa dentro del marco.

### **Documento de Interfaces múltiple con ayuda (MDI frame with MicroHelp)**

Uso. Mismo que la MDI frame.

Características Mismas que la MDI frame

Presenta un renglón de ayuda en línea al usuario

### **Controles**

Son objetos que están puestos en una ventana y que permiten al usuario interactuar con la aplicación. Como los controles son objetos por si mismos, estos tienen atributos y eventos, la mayor parte de la interacción del usuario en la aplicación ocurre a través de los eventos que generan los controles en una ventana.

Algunos controles que usted usa son estándar en windows, y otros específicos de PB como las Datawindows, además se puede crear sus propios controles.

Ejemplos de Controles:

Check Box	MultiLineEdit
CommandButton	UserObject
Datawindow	Picture
DropDownListBox	RadioButton
EditMask	SingleLineEdit
GroupBox	StaticText
HScrollBar	VScrollBar
ListBox	

## **Foco (Focus)**

Identifica en que parte de la pantalla se realizará la siguiente acción.

Para establecer el foco, el usuario puede

Mover el Cursor del Mouse hacia el control y dar un click

Presionar el Tabulador, hasta que el control se selecciona

Un control puede perder el foco sí:

El usuario hace click en otro control

Cuando presiona la tecla Tab.

## **Herencia**

Es una característica que le permite construir una ventana, objeto de usuario y menús que son derivados de un objeto existente.

Ventajas del uso de la herencia

Cuando se cambia un objeto padre, los cambios se reflejan en todos los hijos.

Se tiene consistencia en el código y objetos de la aplicación.

## **Optimizando los descendientes**

Cuando se construye una ventana descendiente se puede

- Cambiar los valores de los atributos y las variables.
- Extender o substituir los scripts.
- Agregar controles.
- Referenciar las funciones, eventos y estructuras de los padres.
- Declarar variables, eventos y estructuras para los descendientes.

## **Instancia**

El Window Painter crea y almacena la definición de una ventana en la librería de PowerBuilder. Esta definición de clase esencialmente llega a ser un nuevo tipo de dato.

Cuando un script abre una ventana llamada window-name, PowerBuilder crea una instancia de una ventana cuya clase es window-name y despliega la ventana.

## Múltiple Instancia

Algunas veces una aplicación necesita tener dos o más instancias de una ventana abiertas a un mismo tiempo (generalmente con datos diferentes).

## Controles de Ventana

Controles de Acción	CommandButton PictureButton
Controles de Estado	CheckBox GroupBox RadioButton
Controles de Despliegue	DataWindow Graph OLE RichText
Controles de Entrada de Datos	DataWindow EditMask MultiLineEdit SingleLineEdit RichText
Control de Etiquetas	Static Text
Controles de listas	DropDownListBox Dropdown PictureBox

ListBox  
 PictureListBox  
 ListView  
 TreeView

Controles de Giro      Horizontal Scrollbar  
 Vertical Scrollbar

Controles de Dibujo      Picture  
 Rectangle  
 Rounded Rectangle  
 Oval  
 Line

Controles de Usuario      User Object  
 OLE  
 Tab

CheckBox :      Enciende o apaga una opción.

CommandButton      Lleva a cabo una acción.

Datawindow      Usado para desplegar manipular y actualizar datos asociados con un objeto Datawindow.

Drawing Object      Usado para mejorar la apariencia de su ventana (line, oval, rectangle, roundrectangle).

DropDownListBox : Permite al usuario seleccionar una palabra de una lista al insertar el nombre en la línea de edición o al dar un click a la flecha del extremo derecho del control para desplegar su lista y entonces dar

click en la selección.

DropDownPictureListBox Igual al anterior pero permite agregar una imagen.

EditMask Como un SingleLineEdit pero provee características especiales de formato.

Graph : Brinda el desplegado gráfico de datos.

GroupBox : Reúne dos o más controles relacionados (generalmente RadioButtons).

HScrollBar and VScrollBar Permiten al usuario indicar cantidades relativas.

ListBox Presenta una lista de varias palabras de texto.

ListView : Muestra la información en forma de lista. Cada palabra consta de un texto y una imagen.

MultiLineEdit Caja utilizada para introducir, editar, y desplegar palabras de texto que puede ocupar más de una línea.

OLE Object Linking and Embedding. Le permite usar otros programas (aplicación servidor) y sus datos en su aplicación. Sus datos permanecen en el servidor de la aplicación en su propio formato, cuando su aplicación incluye un objeto OLE este puede invocar al servidor de aplicación para manipular el objeto. Picture Una imagen bitmap para realizar una acción.

PictureButton : Lo mismo que el CommandButton pero usted puede especificar una imagen. PictureListBox Igual que el ListBox pero cada texto

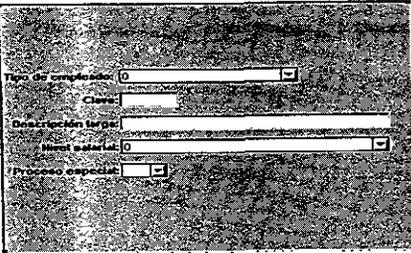
puede incluir una imagen. La imagen puede ser un bitmap, icono o cursor, RadioButton Enciende o apaga una acción mutuamente excluyente.

RichText:	Control que contiene un documento. Muestra el texto formateado y puede contener datos de entrada de un DataWindow control.
SingleLineEdit	Usado para desplegar una línea de texto. StaticText : Usado para encabezados, rótulos y otras literales.
Tab	Control que contiene tab pages en los cuales se colocan controles. Cada tab page tiene su propia etiqueta, imagen y color de fondo.
TreeView	Muestra la información de manera jerárquica. Cada texto del
Treewiew	Control consiste de un texto y de una imagen.
UserObject	Un objeto reutilizable definido por el usuario.

### Algunos controles de una ventana

Se usan los botones de la barra de herramientas dropdown para añadir algunos tipos de control a la ventana. La siguiente lista muestra los botones de la barra de herramientas que pueden emplearse para los diferentes tipos de control y el uso general para cada control.

Control	Descripción
CommandButton 	Ejecuta una acción.

Control	Descripción
CheckBox 	Alterna las opciones prendido, apagado
RadioButton 	Alterna exclusivamente las opciones prendido y apagado
Datawindow control 	Objeto Datawindow Es un objeto que lee y manipula datos de una Base de Datos u otra fuente de datos externo como son archivos de Excel, dBase. Este objeto puede manejar diferentes maneras de la presentación de datos como es por renglones, en forma libre, etiquetas, ediciones periodística. Además se pueden crear campos calculados, imágenes, gráficas
Dropdownlist Box 	Presenta listas predefinidas de elementos para su selección.
User Object	Es un objeto visual, heredado de los objetos del sistema de PowerBuilder, o puede ser un objeto personalizado no visual, en donde se almacenan propiedades y funciones de usuario.

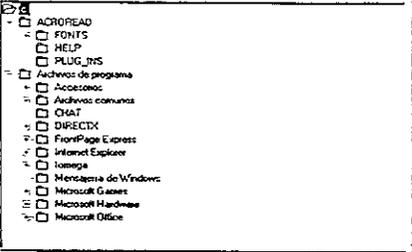
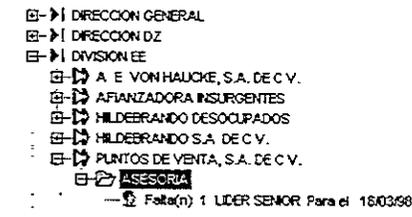
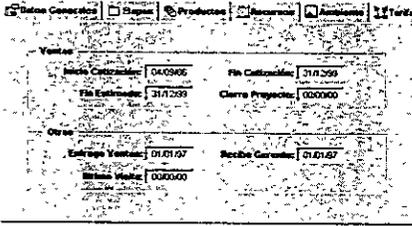
Control	Descripción
<p>TreeView</p> 	<p>Es una forma de presentar los elementos en forma de un árbol, como se muestra en el ejemplo; un directorio que buscando los archivos a medida que se oprime doble click.</p>
<p>ListView</p> 	<p>Muestra una lista de la información del usuario, cada item del listview consiste de un texto y una pintura, la cual puede ser manipulada durante la aplicación. En este caso la lista de item son los faltantes de recursos (Lider senior).</p>
<p>Tab</p> 	<p>Es un tipo de objeto donde se muestra la información ordenada según su tipo, es decir, en una carpeta se ponen los empleados, sueldos, categorías, etc. Como se muestra en la figura.</p>

FIGURA 2.5.11 Resumen de características y beneficios

## 2.5.9 Características Técnicas De Powerbuilder

PowerBuilder es una herramienta de desarrollo que corre en diversas plataformas como es Unix, Macintosh, Window, Window NT, para su instalación se requiere aproximadamente de 70 MB.

El Install Builder puede ayudar a crear instalaciones en discos que contienen los archivos de la aplicación y componentes ejecutables.

## Desventajas

El costo del Software, respecto a otras herramientas de desarrollo, es muy elevado.

## Características :

- Un conjunto de herramientas de desarrollo llamadas painters que proveen una interface gráfica (point - click), para crear ventanas, menús y otros objetos.
- Lenguaje de programación (power script) para crear aplicaciones manejadas por eventos (event - driven).
- Soporta controles hechos para Windows del tipo VBX 1.0 y OCX.
- Un objeto personalizado, llamado datawindows, es usado para la manipulación de la base de datos y reportes.
- Control de versiones y depurador integrado.
- Soporte para intercambio dinámico de datos (DDE), librerías dinámicas de enlace (DLL), y objetos ligados y embebidos (OLE 2.0).
- Soporte para importar y exportar los más comunes formatos de archivos (Dbase, Clipper, Excel, Textos, etc.).
- Extensa ayuda en línea.

## Beneficios :

- Crear rápidamente aplicaciones que utilizan base de datos orientadas a objetos sin tener que recurrir a escribir códigos en C o C++.
- Sacar ventaja de la arquitectura cliente/servidor dos capas, tres capas y multicapas.
- Crear atractivas y fáciles aplicaciones que proporcionen interfaces de usuarios gráficas(GUI) fundamentando el diseño.
- Diseño de aplicaciones que son independientes de un manejador particular de base de datos, usando nativos drivers u open database connectivity(ODBC).
- Creación de objetos reusables.

## 2.6 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DE SQL ANYWHERE

Hoy en día se ha establecido un ambiente de paso rápido, el acceso confiable a la información corporativa es crítica al éxito de un negocio. Para asegurar que los datos estén disponibles cuando y donde se les necesite, requiere de una base de datos de la empresa para salvar y distribuir información. SQL Anywhere ofrece la opción para la empresa pequeña con requerimientos informáticos. Esta solución de gran alcance de la base de datos se optimiza con los requisitos del **e-business** (comercio entre negocios en internet), aplicaciones departamentales y los negocios clasificados como pequeños-medios.

SQL Anywhere ofrece características por completo, base de datos al calibre de la empresa junto con tecnología de gran alcance de sincronización, proporciona una combinación de diseño y desarrollo de acuerdo a las reglas del negocio con facilidad, tiene entrega de ricas funciones en la base de datos para múltiples plataformas, incluyendo Linux, Windows NT, Novell y Solaris. Con su alto rendimiento, optimizador de sentencias SQL. SQL Anywhere ha fijado un nuevo estándar con la cero administración de la base de datos. Y, como la aplicación a las demandas del usuario crecen, también asegura el funcionamiento con un gran número de usuarios en línea y remotos.

### La potencia de la empresa en un DBMS

El pequeño SQL Anywhere, ofrece todo el poder de un DBMS de gran alcance, puede utilizar puesta en práctica de un solo usuario o multiusuarios y además un tratamiento de transacciones completo. Proporciona funciones ricas, incluyendo **Stored Procedures**(procedimientos almacenados), **Triggers**(disparadores de integridad de referencia), bloqueos a nivel registro, recuperación automática de datos, soporta Java. Por lo que se convierte en un recurso eficiente - requiriendo aproximadamente 2 MB de RAM y 4 KB por la conexión del cliente.

---

## **Fácil utilizar, fácil administrar**

Cuenta con auto **self-tuning** (auto afinación) y la administración virtualmente es cero, haciéndolo ideal para las organizaciones con pocos recursos tecnológicos. Un grupo de trabajo o una pequeña empresa puede utilizar la solución sin dedicar a un administrador de la base de datos o a un personal adicional a mantener la base de datos. Además, permite que las complejidades de la transferencia de datos ocurran transparente a los usuarios finales, dándoles el acceso fácil a las funciones de gran alcance. Incluye la herramienta de Sybase Central, una herramienta gráfica de administración que es fácil de utilizar e intuitiva, la administración central de las bases de datos vía remota y todo el ambiente de la replicación.

Sybase Central es bastante directa para los profesionales no técnicos, pero contiene características para satisfacer incluso al profesional más exigente de la base de datos.

## **Escalable**

Soporta un gran número de usuarios en línea y remotos – acorde a las necesidades de demanda o de la aplicación. Por otra parte, SQL Server Anywhere forma parte de Sybase Adaptive Server DBMS, por lo que puede ser escalable a un servidor empresarial o con mayores requerimientos.

## **Soporte a Multiplataforma**

Soporte desde Windows CE, Windows 3.x/95/98, Windows NT y Novell NetWare, Sun Solaris/SPARC, Sun Solaris/Intel, HP UX, IBM AIX, y Linux.

## **Portabilidad Web**

Incluye un servidor abierto, altamente escalable a aplicaciones Web Server que permita mover aplicaciones al Internet para el acceso y la disponibilidad crecientes de información.

## **Estancia Conectada**

Proporciona una sincronización bidireccional de información, mensaje basado en la sincronización optimizada en línea para ambientes de trabajo ocasionalmente conectados. Los usuarios remotos pueden tanto enviar cambios y recibir actualizaciones de la base de datos corporativa - siempre y cuando puedan estar conectados. Porque solamente se actualizan los datos cambiantes y ocurre el proceso de la transacción fuera de línea, los costos asociados con tiempos de conexión se reducen.

## **Conjunto de productos Integrados**

SQL Anywhere esta pensado para desarrollar y diseñar soluciones en cualquier ambiente de trabajo. Las herramientas de gran alcance de desarrollo y de productividad incluidas en SQL Anywhere, permiten el desarrollo de ambientes gráficos, diseño de esquemas de bases de datos o de ingeniería de reversa. Los usuarios también pueden consolidar rápidamente volúmenes de datos grandes personalizados, presentación de reportes con calidad.

## **Soporta dispositivos móviles de la siguiente Generación**

Incluye la opción de desarrollo de tecnología UltraLite y MovilLink Server para Windows CE en plataformas de computo móvil (PalmSize). Sybase extendió el poder de esta nueva tecnología de gran alcance, a la base de datos de Sybase a una nueva clase entera de los dispositivos del handheld.

**Requerimientos de Sistema (Wintel)****Requerimientos del Servidor de BD****PC's basadas en Intel**

- Windows 3.x, Windows 95/98, Windows NT 3.51 o mayor, Novell Netware.
- Intel 80486 o mayor, 8 MB disponibles en RAM

**Dispositivos compatibles con Windows****CE**

- PC con Windows NT 4.0, o Windows 95/98
- Windows CE 2.0 o mayor, 16MB recomendado

**Requerimiento Cliente**

- PC con CD-ROM corriendo Windows 3.x, Windows 95/98, Windows NT 3.51 o mayor

**Soporte en Redes**

- NetBIOS, TCP/IP, Novell NetWare IPX

**MobiLink Server Synchronization****Requerimientos**

- PC con CD-ROM corriendo Windows NT 3.51 o mayor
- Mínimo 2 MB RAM, 5 MB de espacio en disco

**Requerimientos Sistemas (Unix)****Requerimientos del Servidor de BD**

- Sun Solaris SPARC 2.5 o 2.6
- Sun Solaris x86 2.6, HP UX 11.0, IBM AIX 4.2 o 4.3, Red Hat Linux X86 5.1 o 5.2
- 8 MB disponibles en RAM

**Requerimientos Cliente**

- PC compatible con IBM corriendo Windows 3.x, Windows 95/98, o Windows NT 3.51 o mayor, o
- Sun Solaris SPARC 2.5 o 2.6
- Sun Solaris x86 2.6, HP UX 11.0, IBM AIX 4.2 o 4.3, Red Hat Linux X86 5.1 o 5.2
- Unidad de CD-ROM

**Soporte en Redes**

- NetBIOS
- TCP/IP
- Novell NetWare IPX
- IBM AIX 4.2, 4.21, 4.3

## **Desarrollo en Tecnología Ultralite**

### **Requerimientos de Desarrollo**

- PC con CD-ROM corriendo Windows NT 3.51 o mayor

### **Soporte a Dispositivos Cliente**

- Dispositivo corriendo Windows CE 2.0 o mayor, con procesador MIPS o SH3
- Dispositivo corriendo en plataforma Palm Computing 2.0 o mayor

# Capítulo 3

## Planteamiento del Problema y Propuesta de Solución

### 3.1 PLANTEAMIENTO DEL PROBLEMA

#### 3.1.1 Situación Actual

La empresa se ve enfrentada a una serie de procesos que describen cual es la situación actual de una empresa consultora.

Una vez que una empresa quiere adquirir los servicios que ofrece una empresa consultora, tendrá una primera cita con un ejecutivo de cuenta de la consultora, durante dicha cita se conoce qué es lo quiere el cliente, o al menos se da una idea de sus requerimientos; muchas veces el cliente no tiene mucha idea de los sistemas de información, por esta razón no sabe describir qué es lo que requiere con exactitud. Si después de una primer cita no se llega a un acuerdo, entonces habrá una serie de citas posteriores según sea necesario, hasta obtener una idea más clara de los requerimientos del cliente. En ocasiones gente del área de operaciones se ve involucrada en cada una de estas entrevistas, tal vez sea necesario que la empresa preste un servicio de consultoría (como reingeniería de procesos de negocio, o implementación de procesos de negocio o plan de negocios) para llegar a un acuerdo.

Después de que se ha llegado a un primer acuerdo con el cliente, la empresa consultora se ve obligada a hacer una propuesta; en dicha propuesta se tratan asuntos financieros, operativos y de tiempo, cabe mencionar que muchas veces el cliente es el que determina dicha propuesta, de cualquier manera, ya sea por parte del cliente o por parte de la empresa consultora, siempre hay una propuesta.

Cuando se haya aceptado la propuesta o en el proceso de aceptarla, se pone en camino el recabar la información que se necesita para que se pueda tomar en cuenta el proyecto, en primeras instancias es necesario obtener la siguiente información:

- Cliente que solicita los servicios
- Nombre del proyecto
- Tipo de proyecto
- Alcance del proyecto (Hasta donde llegara su funcionamiento)

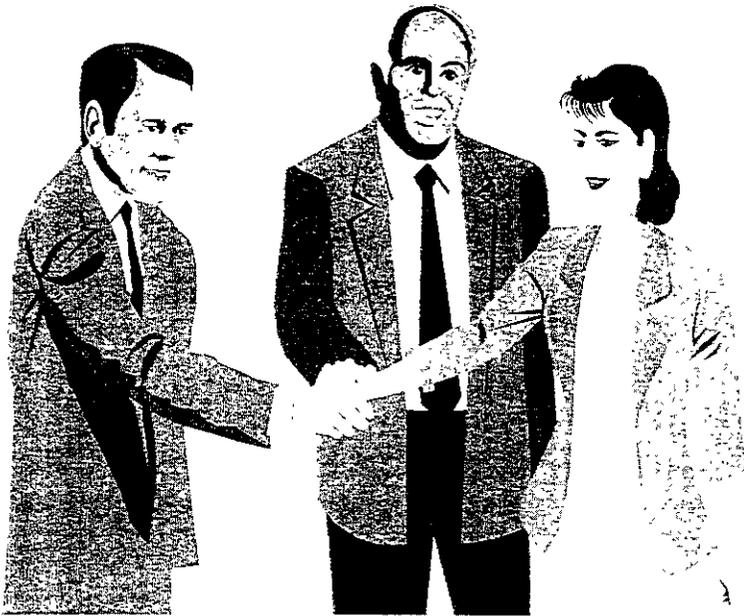


FIGURA 3.1.1 Llegan a un acuerdo común el cliente y la empresa consultora.

- Dirección del cliente
- Fecha que se quiere inicie
- Fecha propuesta para terminar el proyecto (no siempre es dato)
- Nombre y teléfono de la persona que hace el trato con el ejecutivo de cuenta
- Nombre y teléfono de la persona que representa el área de sistemas del cliente, si

es que existe

- Nombre y teléfono de la persona que representa al usuario del sistema
- Nombre y teléfono de la persona con quien se arregla la factura
- Fax del cliente
- Una descripción del proyecto, es decir la problemática a tratar por el proyecto
- Director divisional de la empresa consultora

Cabe mencionar que en el punto, tipo de proyecto, nos referimos en cierta manera al giro del proyecto, es decir, esto podría ser:

- Administrativo
- Financiero
- Educativo
- Soporte técnico
- Una Capacitación

O algún otro, una vez que la propuesta sea totalmente aceptada, la empresa consultora, necesitará información adicional para que se pueda iniciar dicho proyecto, dicha información es básica para el inicio mencionado, como:

- Etapas del proyecto
- Productos a entregar
- Recursos (personal requerido)
- Tecnología y herramientas a utilizar
- Un plan de trabajo

A continuación definimos de qué trata cada uno de los puntos anteriores.

**Etapas del proyecto.** Se necesita definir de qué etapas constará el proyecto, no necesariamente es una de las siguientes, aunque si son procesos que están ya definidos.

- Análisis
- Diseño
- Desarrollo
- Implantación
- Mantenimiento
- Liberación
- Desarrollo de ayudas en línea
- Prototipo
- Pruebas Modulares
- Pruebas integrales

Cabe mencionar que a cada etapa se le asigna una fecha de inicio y una de fin, aunque como es bien sabido, en el área de desarrollo de sistemas las fechas podrían ser cambiadas.

**Productos a Entregar.** La empresa consultora se compromete a entregar una serie de documentos o artículos al cliente, estos podrían ser los siguientes:

- Carpeta de análisis
- Carpeta de diseño
- Manual de usuario
- Manual técnico
- Programas fuente
- Discos de instalación

Es importante mencionar que tal vez sea necesario especificar ciertos comentarios del por qué o para qué, de cada producto entregable.

**Recursos (personal requerido)** Para que la consultora pueda efectuar un buen servicio, necesita contar con el personal calificado que sea capaz de efectuar los requerimientos necesarios, la manera como las empresas consultoras designan la

experiencia o conocimientos de su personal es, dándoles un perfil o "puesto", es decir, adjudicando jerarquías a su personal. Estos puestos o jerarquías son descritos frecuentemente de la siguiente manera:

- Analista programador junior.
- Analista programador senior
- Ingeniero de sistemas junior
- Ingeniero de sistemas senior
- Ingeniero especialista
- Líder de proyectos junior
- Líder de proyectos senior
- Becario
- Consultor

Y existen otros nombramientos, que dependen de la empresa consultora.

La manera como el cliente solicitará los requerimientos para su proyecto; será en base a los puestos del personal de la empresa consultora, por ejemplo, para un proyecto se pueden necesitar un líder de proyecto, cinco analistas programadores Jr., 2 analistas programadores Senior y un ingeniero de sistemas Jr.

**Tecnología y herramientas a utilizar.** Dependiendo del proyecto y de común acuerdo, el cliente junto con la consultoría, seleccionan las tecnologías y herramientas de software y hardware a utilizar.

El personal requerido junto con las tecnologías y herramientas a utilizar, darán la pauta para seleccionar al personal que más se adecue a dichos requerimientos, la consultora se ve enfrentada a una toma de decisiones que tal vez se complique demasiado, las razones son muchas, tal vez no se cuente con la gente que conozca el software que requiere el cliente, o tal vez el personal que conoce el software esta ocupado en otro proyecto, o quizás existe quien conozca el software pero no cumple con ser un líder de

sistemas (suponiendo que la empresa no quiere otro puesto), o quizás convenga sacar a un recurso de un proyecto para poder asignarlo y después cubrir el puesto de éste; además es importante mencionar que el proyecto también exige un cierto nivel de experiencia en las herramientas; siendo la experiencia un factor importante para la toma de decisiones, lo cual será un tema que se tratará adelante con más detenimiento. De cualquier forma ésta también influye en el personal que se vaya a elegir; en fin las probabilidades son muchas. De cualquier forma la consultora tendrá que cubrir los requerimientos del cliente, si no, puede significar desde pérdida de dinero y tiempo, hasta pérdida del proyecto o aun más fuerte, perder al cliente.

Hoy en día, hablar de tecnologías y herramientas es hablar de toda una gama de conocimientos, cada día se ve actualizado el mundo de la informática y las comunicaciones, es por eso que la consultora se ve forzada a actualizarse cada día. Como información, hacemos mención de algunas tecnologías y herramientas que existen hoy en el mercado.

#### Manejadores de bases de datos

- SQL Anywhere
- Access
- Informix on\_line 7.0
- Sybase

#### Lenguajes de alto nivel

- Cobol
- C
- C++
- RPG/400
- Informix ISQL\_C
- Algol

- JCL

#### Front end gráfico

- Power Builder
- Visual Basic
- Delphi
- New era
- Developer 2000
- SQL Windows
- Centura Team
- Visual C++

#### Case y/o metodología

- Erwin
- Designer 2000
- Data ware House
- System Architect
- S Designer

#### Comunicaciones e Internet

- FTP
- Tuxedo
- ODBC 16 y 32 bits
- TCP/IP
- IPX
- Módems
- Redes Lan

**Plan de trabajo.** El establecimiento de tiempos para cada etapa del proyecto es información que deberá de conocer todo el personal que esté involucrado en el proyecto. Es importante establecer fecha de inicio y finalización de cada etapa, como ya habíamos mencionado, pero también será importante, como información, dar fechas de inicio de la cotización y fin de esta, fecha de cierre de proyecto, fecha estimada de fin de proyecto, fecha de la última visita que hizo al cliente el jefe divisional.

Una vez que el proyecto es puesto en marcha, es necesario llevar un seguimiento para poder evaluarlo, tanto al proyecto en general como a cada una de las personas que intervienen en éste, y de esta forma poder satisfacer las necesidades que se vayan presentando, o tal vez será bueno evaluar el por qué se canceló o no se autorizó.

Para la evaluación del proyecto, será necesario saber cual es el estado del proyecto en cuanto a su situación; estamos hablando de que podría estar en alguna de las siguientes situaciones, o en alguna otra.

- Prospecto
- Negociación comercial
- Autorizado
- En desarrollo
- Suspendido momentáneamente
- Liberado por operaciones
- Concluido
- En trámites finales con ventas

Basándose en uno de los estados anteriores y el desempeño a grandes rasgos del proyecto podemos evaluarlo, es decir de alguna manera saber cual es su situación. Hablando de resultados y tiempos, podríamos hablar de las siguientes situaciones:

- En observación
- Sin evaluación

- Con problemas
- Critico
- Estable
- Retrasado

Basados en esta evaluación, es bueno saber el motivo por el cual se califica de esa manera a un cierto proyecto, y también más importante todavía qué acciones se van a tomar para darle solución a un proyecto que tiene problemas.

La mejor manera de evaluarlo es, primero a manera general, y después a nivel área, esto es importante porque tal vez el desempeño del proyecto está teniendo problemas debido a que el área de recursos humanos no ha conseguido el personal necesario, en ese caso el problema se canaliza a dicha área, o quizá al área financiera no pagó a tiempo a un empleado, e hizo que este prefiriera renunciar, esto causa que el principal problema lo esté teniendo dicha área.

Para una mejor evaluación hay una serie de cuestionarios al cliente durante y al final de la entrega del proyecto, y también a la empresa consultora con respecto al cliente.

Respecto a la evaluación del personal, es también importante llevar un seguimiento de las actividades de cada persona durante el transcurso del proyecto, tomar en cuenta su experiencia y desempeño tanto humano (de actitud), como de la tecnología y herramientas que maneja durante el proyecto; para esto será necesario evaluar el nivel de conocimiento que tenga en cierta herramienta, algunos niveles de conocimiento podrían ser los siguientes:

- Intermedio
- Básico
- Experto
- Avanzado
- Principiante

Pero como ya se había mencionado, también será importante el desempeño que presente el individuo, podríamos definir este desempeño con alguno de los siguientes calificativos:

- Mínimo
- Bueno
- Muy Bueno
- Excelente
- Deficiente
- Nulo

Aunque también hay que tomar en cuenta que no toda la experiencia de un empleado se adquiere por medio de un proyecto; habrá ocasiones que la fuente que nos informe acerca de su nivel de conocimiento y desempeño de alguna herramienta sea su curriculum u otra fuente, por esta razón también es importante saber la fuente de la cual se obtuvo su evaluación, podríamos sugerir las siguientes:

- Evaluación del cliente
- Examen
- Curriculum
- Curso recibido
- Curso impartido
- Certificación
- Aprendizaje durante el proyecto

Es importante que la empresa lleve un seguimiento de la evaluación de su personal, porque de esto depende quién puede ser asignado a un proyecto, con estos factores podremos asegurar un mayor éxito al proyecto y por consiguiente a la empresa, además de que es información básica para la toma de decisiones.

### 3.1.2 El Usuario y sus Requerimientos

Al realizar cualquier sistema de información, quizá el usuario es el factor que más influye tanto en la vista como en los procesos internos de dicho sistema, muchas veces al enfrentarnos con el usuario es necesario ayudarlo a que pueda expresar exactamente qué quiere que pase al utilizar el sistema, u otras veces es necesario sugerirle ideas. De cualquier forma la opinión del usuario siempre es importante, por esta razón es necesario considerar los siguientes puntos:

1. Identificar plenamente las necesidades del cliente (usuarios). Es necesario tener una visión clara de lo que el cliente tiene en mente, acerca de lo que el sistema le puede ofrecer y de lo que él necesita. Esto se lleva a cabo entrevistando al cliente, marcando los objetivos del sistema, por ejemplo, la información que se introducirá al sistema, y la información que el sistema generará, las funciones que los usuarios tendrán, el rendimiento que se espera del sistema etcétera. Este punto no será cubierto sin una comunicación efectiva entre el cliente y el desarrollador.

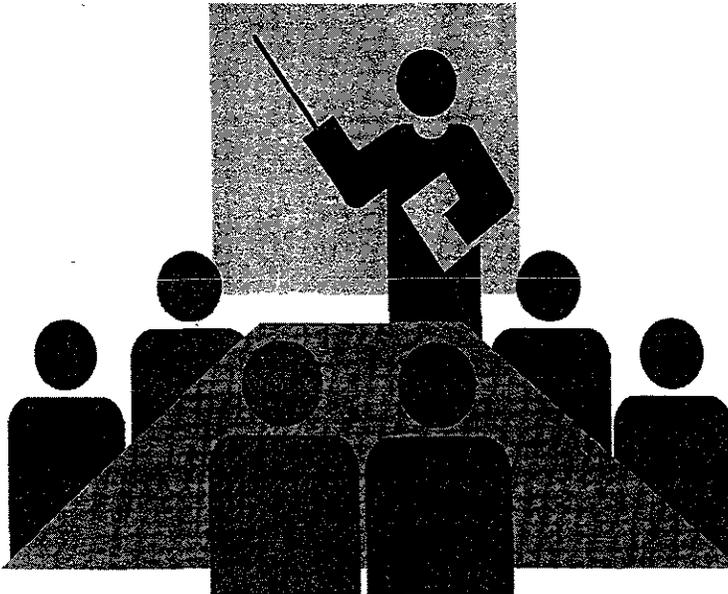


FIGURA. 3.1.2.1 Entrevista con el cliente

2. Estudiar la viabilidad del proyecto. Esto es si el sistema puede ser llevado a cabo, muchos proyectos se han dejado inconclusos porque no eran viables. La viabilidad del proyecto puede dividirse en cuatro partes:
  - Económica. Se refiere a la evaluación del costo al momento del desarrollo comparado con el beneficio que el sistema le brindará al cliente.
  - Técnica. Aquí se analizan las restricciones que pueden afectar la posibilidad de realizar un sistema que cumpla con las expectativas del cliente, y estas restricciones son de funcionalidad y rendimiento.
  - Legal. Todo elemento que forme parte de una sociedad, se ve obligado a respetar las leyes establecidas, y el campo de la computación es la excepción.
  - Opciones. En la mayoría de los casos, siempre se pueden encontrar varias opciones para enfrentar una determinada situación, y el estudiar estas nos brinda la oportunidad de elegir la mejor.
  
3. Realización de un estudio técnico y económico. Hoy en día el mundo de la computación cambia aceleradamente, con lo cual un buen estudio técnico del sistema nos podrá respaldar ante estos cambios. Con respecto a lo económico hay que enfatizar que para la mayoría de las empresas en las cuales su objetivo es aumentar sus ganancias, este puede ser un punto crítico para el éxito o fracaso del proyecto.
  
4. Identificar las funciones que cada una de las partes que van a actuar en el sistema realizarán. Hay que determinar que funciones específicas van a realizar el software, el hardware, los usuarios, la base de datos y en sí todos los elementos del sistema.
  
5. Establecimiento de costos y tiempos
  
6. Definición del sistema para el trabajo de ingeniería que se comenzará a realizar.

Tomando en cuenta los puntos anteriores, encontramos que el usuario requería lo siguiente:

Respecto a la interface del sistema.

- Debe ser funcional, agradable a la vista. Cabe mencionar que para poder lograr una interface que cubra esta expectativa se deben entender a fondo los requerimientos de la empresa y se debe de tener un amplio criterio de la estética y el orden.
- Debe de ser organizado y con un comportamiento coherente.
- La captura de datos debe ser en extremo ágil a fin de poder capturar en el menor tiempo posible toda la información recibida
- Debe de permitir todo tipo de consultas a la información bajo una amplia gama de criterios.
- Debe ser fácil de manejar, que al ver las pantallas se intuya cuales serán los pasos a seguir.
- Una interface gráfica bien definida, es decir, que dependiendo de las funciones que realicen los objetos que tengan gráficos, los gráficos describan o den idea de la función del objeto.
- Una pantalla especial de reportes que nos permita imprimir, cambiar impresora, ver el número de página en el que estamos, número de páginas de que consta el reporte, poder hacer zoom (acercamiento a los elementos) en el reporte, poder escoger impresora sin salir de la aplicación.
- En la información que lo requiera, poder ordenar los datos bajo los criterios más importantes.
- Los mensajes que el sistema proporcione al usuario deben de ser claros y breves y sólo cuando sean necesarios.

Respecto a otros requerimientos del usuario, se obtuvieron los siguientes puntos.

- La pantalla de captura de proyectos, debe mostrar los datos más significativos, además de los elementos relacionados con éste, como plan de trabajo, etapas, tecnologías y herramientas, productos entregables, equipos de trabajo.
- El sistema contendrá consultas a los equipos de trabajo en los proyectos, de tal forma que proporcione el perfil del empleado en el proyecto, que no necesariamente es el requerido.
- Presentar una pantalla para la evaluación del personal
- Presentar una pantalla para la evaluación del proyecto
- Presentar las pantallas necesarias para la evaluación del proyecto cuando este finalice, recordando que dicha evaluación se basará en cuestionarios proporcionados por el usuario.
- Información suficiente para la toma de decisiones a la hora de escoger el personal que será asignado a un proyecto, como el nombramiento de la persona, su experiencia y desempeño, su historial de proyectos en los que ha participado; si está como recurso desocupado, el proyecto en el que se encuentra actualmente, si es que está asignado a un proyecto cuando se desocupará, curriculum del empleado como un reporte, que nos permita ver qué empleados cumplen con las condiciones del cliente.
- Presentar reportes que permitan ver los proyectos con su información, la experiencia del personal, las evaluaciones del proyecto tanto de manera general, como por áreas, reportes de la evaluación de proyectos cuando estos finalizan, información general de los empleados.

- Todos los catálogos que sean necesarios para el mejor manejo de la información y de la captura.
- Seguridad para entrar al sistema, para las opciones de menú, y hasta para los objetos de las ventanas con el fin de que personas ajenas a la información se abstengan de intentar verla o modificarla.

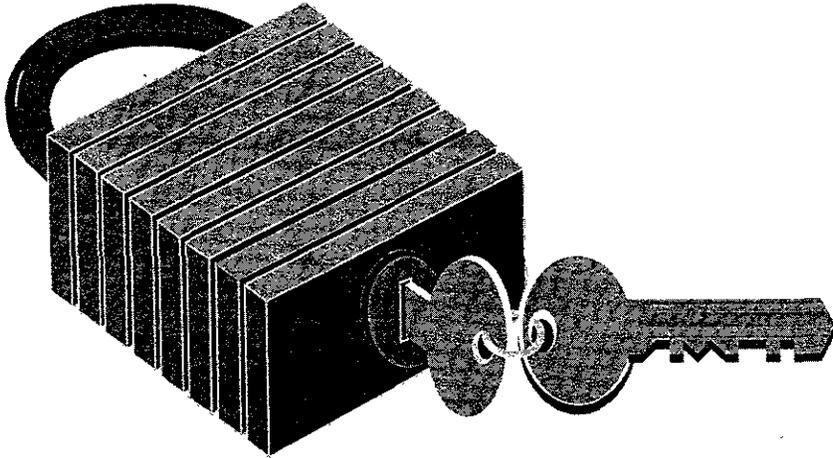


FIGURA. 3.1.2.2 La seguridad en un sistema hoy en día es parte necesaria

- El sistema debe de contar con manuales, de tal forma que estos proporcionen la información necesaria para el manejo del sistema, de manera rápida y fácil.

### 3.1.3 Búsqueda y Análisis de la Información

En la toma de decisiones de una empresa siempre es necesario contar con la información adecuada, de ahí la importancia que se obtenga de las fuentes correctas y que además podamos asegurar que es real.

Para que se pueda hacer un análisis óptimo de la información es necesario saber cuándo determinada información dependerá de otra, o cuando es el momento para que sea tomada en cuenta, es decir no se puede por ejemplo determinar cuáles serán las

etapas de un proyecto si antes no se ha hablado con el cliente.

En las consultorías siempre que un proyecto inicia, es necesario tener en cuenta cada aspecto que involucre el proyecto, ya que, si no es así la consultora puede perder tiempo y dinero.

Hace algún tiempo en cierta empresa consultora se negociaba un proyecto; el proyecto era para la secretaría de hacienda y crédito público, dicho proyecto implicaba el concepto de programación en tres capas; la tecnología y herramientas que implicaba el proyecto requería de conceptos de tecnología muy avanzados y algunos de ellos todavía no muy conocidos en México, el proyecto de acuerdo al trato inicial iba a tener una primer fase que duraría aproximadamente un año y la cantidad de personal necesario para dicha fase rebasaba las ciento cincuenta personas, como la empresa no contaba con todos estos recursos disponibles, comenzó a comprar equipo, contratar una gran cantidad de personal y a sacar gente muy capaz de otros proyectos, sin saber aún que tan viable era el proyecto, es decir faltaba información para determinar esto.

Lamentablemente la consultora se confió y siguió con las contrataciones, después de casi tres meses de análisis de la información, se concluyó que el proyecto todavía no era viable con lo que se encontraba disponible en ese momento. Debido al retraso y a la falta de recursos para ciertas herramientas, la consultora necesitaba mas tiempo, y el cliente se comenzó a desesperar, al final el proyecto no fue autorizado, la consultora perdió una gran cantidad de dinero y tuvo que despedir a la gente que había contratado y regresar al personal que había movido de su proyecto original.

Para que la empresa consultora considere un proyecto, necesita tomar en cuenta la información inicial necesaria, dicha información es recabada por el ejecutivo de cuenta junto con el cliente, esto es, datos personales del cliente y algunos datos iniciales del proyecto como la descripción y el tipo de proyecto, la lista completa de los datos iniciales se encuentra en el tema 3.1, como ya también se había mencionado, tal vez sea necesario una serie de citas posteriores con el cliente, en ocasiones en compañía

de uno o más consultores enviados por el área de operaciones. En las citas posteriores se obtiene la información complementaria del proyecto, dicha información es.

- Etapas del proyecto
- Productos a entregar
- Recursos (personal requerido)
- Tecnología y herramientas a utilizar
- Un plan de trabajo



**FIGURA 3.1.3.1 El ejecutivo de cuenta recaba la información inicial del proyecto**

Cabe mencionar que para determinar las etapas del proyecto es necesario que gente con la experiencia necesaria se entregue a dicha tarea (gente del área de operaciones), esta gente junto con el cliente deciden de cuántas y cuáles etapas constará el proyecto; el recabar dicha información puede llevar de unas horas a varios días, o en ocasiones cuando el proyecto es muy grande hasta semanas, ya que la descomposición en etapas debe también de contemplar tiempos. Para darle el tiempo a una etapa casi siempre es necesario descomponerla en módulos, y asignar cada modulo a la persona adecuada, porque no puede haber retrasos. En ocasiones, dada la complejidad de ciertas etapas o porque aún no estén bien definidas, sólo se define el

nombre de la etapa y se dan fechas de inicio y finalización estimadas, para después definir de manera correcta los tiempos, aunque a veces también se da la necesidad de aumentar ciertas etapas durante el desarrollo del proyecto.

La manera como se definen los productos a entregar, es nuevamente entre el cliente y el representante del área comercial, y si es necesario con algún consultor del área de operaciones; una vez definidas las etapas, el cliente decidirá qué quiere que se le entregue, hay que tomar en cuenta que cada uno de estos productos influye en el costo del proyecto, y no siempre el cliente los exige, la entrega de estos productos casi siempre se va dando al termino de cada etapa.

Para definir el personal requerido, se maneja el mismo procedimiento, el cliente llega a un acuerdo junto con el vendedor, y deciden qué tipo de recursos o puestos se necesitarán para el desarrollo de cada una de las etapas, el recabar el personal adecuado para integrar el equipo de trabajo necesita de una gran cantidad de información, como ya se había mencionado, el recabar la información para determinar quién será asignado al equipo de trabajo, es un trabajo muy pesado, él área de recursos humanos da el curriculum de cada empleado, y los proyectos en los que ha participado cada persona dentro de la consultora, el área de operaciones conoce los equipos de trabajo asignados a cada proyecto y hasta cierto punto, el nivel de conocimientos que cada empleado tiene en cada herramienta y la actitud ante su cargo. Recursos humanos junto con operaciones determinan cuando un recurso asignado a un proyecto será liberado. recursos humanos lleva una lista del personal que se encuentra desocupado en ese momento; toda esta información nos lleva a hacer un análisis exhaustivo para determinar el equipo de trabajo.



El cliente pide los recursos al ejecutivo de cuenta o el mismo ejecutivo los sugiere



La lista de recursos llega al área de operaciones



Recursos Humanos junto con operaciones hacen una propuesta de posibles candidatos

**FIGURA 3.1.3.2** Secuencia para determinar el equipo de trabajo de un proyecto

El uso de la tecnología y herramientas a utilizar durante cada etapa del proyecto es algo que se decide también entre el cliente, el área comercial y el área de operaciones

dependiendo de las necesidades del proyecto, en ocasiones el cliente decide totalmente el sistema operativo y el lenguaje o el manejador de bases de datos o cualquier otra tecnología junto con sus herramientas, pero habrá ocasiones en que se le tenga que sugerir y dar argumentos del por qué de la sugerencia; en otras ocasiones el cliente deja que la empresa consultora decida totalmente, siempre y cuando el costo de la tecnología sea aceptable, en otras ocasiones el cliente junto con los representantes de la consultora hacen un análisis, evaluando pros y contras de las herramientas a escoger, de cualquier forma se llega a un acuerdo entre el cliente y la empresa consultora.

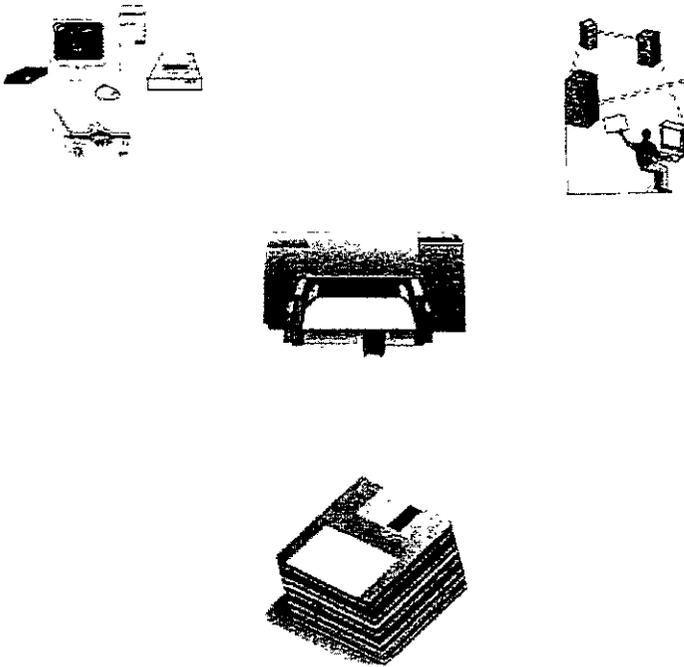


FIGURA 3.1.3.3 El uso de la tecnología y herramientas de software y hardware a utilizar en el proyecto la deciden entre el cliente, el área comercial y operaciones

Respecto al plan de trabajo, después de haber planeado los módulos de cada etapa podemos decidir que tiempo llevara el proyecto en total o por lo menos dar una fecha aproximada. Cabe mencionar que los tiempos dependen muchas veces de cuanto personal se asigne a cada etapa, dichos tiempos son planeados por gente del área de operaciones prácticamente, aunque en ocasiones especiales el cliente decide tiempos, esto se llega a dar cuando el cliente necesita el sistema para una determinada fecha, en esos casos la consultora se ve forzada a cumplir mientras sea factible, quizá será necesario asignar una mayor cantidad de recursos, o simplificar algunos módulos, lo que sea necesario para terminar en el tiempo pedido. Muchos tiempos y fechas se van dando durante el transcurso del proyecto.

La información para evaluar un proyecto, la podemos obtener de los resultados del equipo de trabajo, de la evaluación del mismo equipo de trabajo, del tiempo transcurrido comparado con las fechas establecidas con el plan de trabajo, de las quejas presentadas, de la satisfacción del cliente, de los problemas que se han presentado, etcétera. Toda esta información puede ser recabada para que podamos evaluar al proyecto desde diferentes perspectivas, incluyendo la evaluación por áreas que ya se había descrito; es importante mencionar que para dar una evaluación correcta al proyecto, debe haber una retroalimentación entre el cliente, el área de operaciones, el área de recursos humanos y el área comercial; además esta evaluación debe ser periódica, siendo los periodos entre cada evaluación aproximadamente de una semana.

Respecto a la evaluación del personal, necesitamos recabar información con el cliente, de los módulos que haga cada persona, de su curriculum, de la actitud que tenga en cuanto a forma de ser se refiere ante determinadas situaciones, de los errores que contengan sus módulos, de sus comentarios en su parte de código, de su capacidad para resolver problemas, etcétera. Como en la evaluación del proyecto también es necesaria una retroalimentación entre el cliente, el área de operaciones, el área de recursos humanos y el área comercial; tal vez esta evaluación no sea tan repetitiva como la del proyecto, pero debe existir siempre que sea necesario.

La evaluación al final del proyecto se consigue por medio de cuestionarios al cliente, y a la misma empresa consultora. Es importante no sólo evaluar al proyecto durante el transcurso de este, sino también una vez que ha terminado, para concluir las evaluaciones de la manera más exacta.

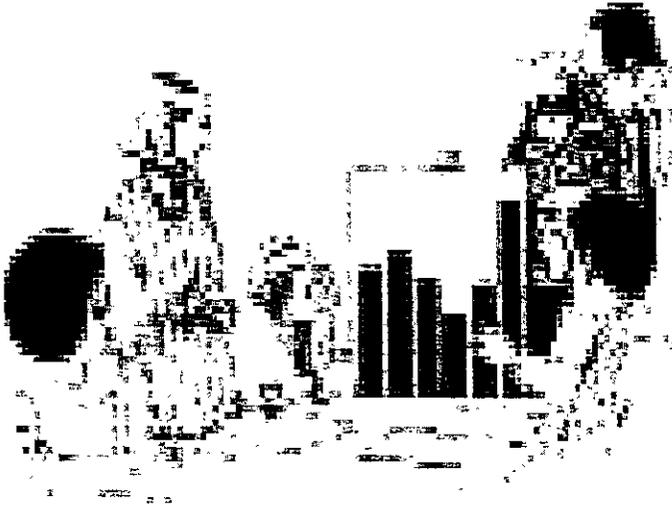


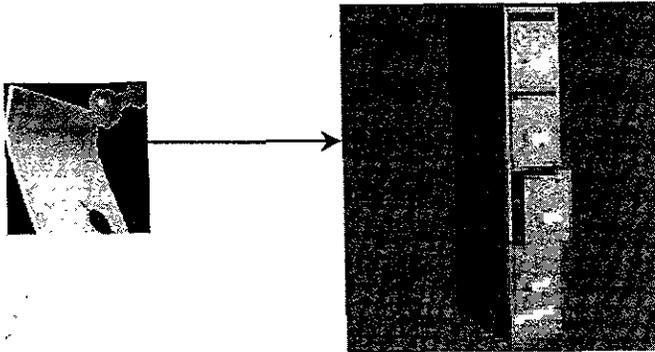
FIGURA 3.1.3.4 La evaluación a proyectos y a personal se da por medio del análisis de resultados

### 3.1.4 Identificación del Problema

Hoy en día las empresas consultoras necesitan dar respuestas rápidas y confiables dada la demanda del mercado y la competencia por el mercado, una empresa consultora que no cuenta con un sistema bien definido para la toma de decisiones no representa una empresa confiable. Actualmente en el mercado Mexicano no existe un sistema integrado que controle la administración de proyectos de las empresas de Consultoría.

Debido a la gran cantidad de información que llevan a cabo las consultorías es difícil llevar un control, nos encontramos ante varios problemas, cada área necesita estar actualizando sus archivos en los archiveros y muchas veces ni siquiera se registra

información que en un futuro podría servir para la toma de decisiones, debido a que la mayor parte de la información se encuentra registrada sobre hojas de papel es difícil y tardado hacer consultas a dicha información, aun la información que se tenga respaldada en hojas de cálculo, la mayoría de las veces no da el manejo correcto para la toma de decisiones. A continuación explicamos con más detenimiento cuáles son los problemas a los que se enfrenta una consultora con la información y toma de decisiones.



**FIGURA 3.1.4.1** Cada área actualiza sus archivos por separado

Una vez que se recabaron los datos iniciales de un proyecto se pasan a un formato que maneja la empresa, por lo regular cada área maneja esos datos bajo ese mismo formato pero cada quién lo llena por separado, esto hace que la información sea duplicada. De la misma forma se llena un formato especificando las etapas del proyecto junto con sus fechas de inicio y fin, dicho formato lleva un espacio para comentarios que se quieran hacer de cada etapa, si es que se requiere. También existe un formato que especifica los productos a entregar con un espacio para hacer comentarios, y otro formato mas para especificar las tecnologías y herramientas a utilizar durante el proyecto, en ocasiones se especifica el nivel de conocimientos que se requiere para el manejo de las herramientas.

De la misma manera se tiene un formato para que el área de recursos humanos se entere del personal requerido, donde se informa el perfil o puesto requerido y la cantidad de recursos solicitados, muchas veces como se había mencionado se especifica el nivel de conocimientos que se requiere que maneje el recurso.

Teniendo toda esta cantidad de información para cada proyecto, y tomando en cuenta como ya también antes se había mencionado, que existen empresas consultoras que están manejando arriba de quinientos proyectos a la vez, se enfrentan a un gran problema, a la hora de consultar la información de cada proyecto, son muchos los archivos que se manejan y la información por proyecto se extiende demasiado, de manera que el acceso a dicha información puede ser lento y no siempre confiable ya que cada área llena los formatos por separado.

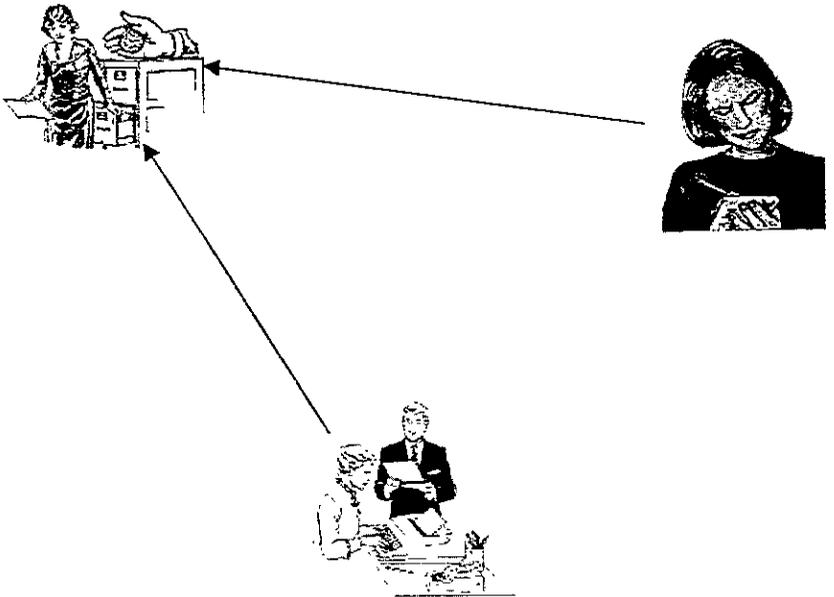


FIGURA 3.1.4.2 La mayor parte de la información es recabada sobre formatos preimpresos

Pero tal vez el principal problema al que se ven enfrentados es el tener que decidir el equipo de trabajo para un proyecto que se inicia o que está en desarrollo (por ejemplo

cuando hay gente que abandona el proyecto o se abre una nueva etapa, o se dan cuenta que se necesitan mas recursos), primero que nada necesitan saber quién se encuentra desocupado en ese momento (no está asignado a ningún proyecto), qué herramientas maneja la gente desocupada y qué nivel de conocimientos tiene de las herramientas, qué personal está a punto de desocuparse de un proyecto y también las herramientas que maneja junto con el nivel de conocimientos que tiene de estas, qué tanta experiencia y el puesto o nombramiento que tienen tanto los desocupados como los que están a punto de desocuparse, su actitud para enfrentar problemas y curriculum tanto dentro como fuera de la empresa, además de otros datos personales secundarios. Una vez que se haya recabado esta información, que por supuesto es bastante tedioso, podrán tomar decisiones para elegir el equipo de trabajo, cabe mencionar que para recabar toda esta información es un trabajo de muchas horas, y por lo regular no se tiene información real y actualizada de los conocimientos reales de cada empleado, además en el caso de que la información no sea real se estarán tomando decisiones bajo riesgo para la empresa consultora, es importante mencionar que la información mencionada se lleva sobre hojas de papel o simplemente ni siquiera se tiene registrada. A la hora de mandar un equipo de trabajo con un cliente hay que tomar en cuenta que el cliente exige respuestas rápidas y validas a sus requerimientos.

De la misma manera también la evaluación de proyectos representa un serio problema, estas evaluaciones también son recabadas en formatos sobre hojas de papel; en las consultoras ocurre muy a menudo que se hagan llamadas del cliente a la compañía consultora, por lo regular dichas llamadas van enfocadas a los sistemas que ya fueron liberados pero que requieren algunas correcciones de errores que no habían ocurrido, o algunas adiciones a ciertos módulos, en general a darle mantenimiento, en ese momento, la persona que atiende la llamada por parte de la consultora necesita saber quien formaba parte del equipo de trabajo, cuales fueron las evaluaciones del cliente hacía el proyecto, cuando se termino y otra información igual de importante para tener un marco de referencia mas amplio y poder dar respuestas mas certeras al cliente, en un momento como ese, se necesita recabar la información rápidamente, pero debido a la forma como se lleva dicha información el proceso es lento.

Por lo regular mientras se desarrolla un proyecto, las evaluaciones a dicho proyecto son la base para tomar decisiones, casi siempre las consultoras hacen juntas cada semana para evaluar todos los aspectos del proyecto, en dichas juntas cada área presenta un informe, además de que las personas encargadas de cada proyecto presentan también un reporte de lo que ha ocurrido durante la semana, otra parte de la evaluación viene también por parte del cliente, en dicha evaluación pueden intervenir tanto usuarios como el jefe de sistemas por parte del cliente; una vez que se ha recabado toda esa información se evalúa el proyecto, nuevamente la carga de papeles con la información se hace demasiado pesada, por lo regular se atienden problemas de semanas pesadas, esto hace que también se necesite saber cuales habían sido los problemas anteriores y que se hizo para solucionarlos, al manejar la consultora una gran cantidad de proyectos, es difícil llevar un control de toda esta información, casi siempre dichas juntas suelen prolongarse bastantes horas y en ocasiones se tienen que posponer hasta varios días después.

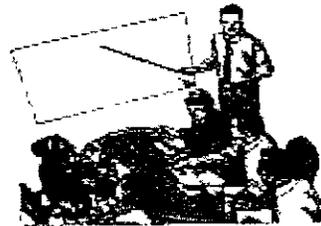


FIGURA 3.1.4.3 Las juntas de evaluación de proyectos se llegan a prolongar demasiado

### 3.1.5 Áreas Involucradas en el Proceso

Desde el inicio, desarrollo y hasta el final del proyecto cada una de las áreas de la empresa, junto con el cliente que se atiende juegan un papel importante en cada una de las partes del proyecto. Aún después de que se concluye el proyecto, es necesario que los proyectos pasen por una etapa de garantía que la empresa consultora da a cada uno de los proyectos que libera.

A manera de resumen a continuación presentamos las funciones principales que cada área o departamento realiza.

#### Operaciones:

- Planeación y desarrollo de la propuesta del proyecto
- Sugiere etapas del proyecto
- Plan de productos a entregar
- Intervención en el plan de asignación de recursos solicitados al proyecto
- Propuesta de tecnologías y herramientas a utilizar durante el proyecto
- Propuesta de un plan de trabajo
- Evaluación a proyectos
- Evaluación a personal
- Desarrollo del proyecto
- Atención al cliente
- Evaluación del cliente
- Supervisión del proyecto
- Asignación del personal al proyecto
- Capacitación al personal de la empresa cliente (si se requiere)

#### Recursos Humanos:

- Evaluación a proyectos

- Evaluación a personal
- Intervención en el plan de asignación de recursos solicitados al proyecto
- Reclutamiento del personal
- Capacitación al personal en estándares manejados, formatos y estilos
- Valuación de puestos
- Altas de personal
- Bajas de personal
- Vacaciones del personal

#### **Área Comercial:**

- Promoción y venta de los servicios de la consultora
- Presentación y concertación de citas con el cliente
- Propuesta comercial del proyecto
- Propuesta comercial de productos a entregar
- Propuesta de valuación de los recursos solicitados al proyecto
- Informe de valuación e información de las tecnologías y herramientas a utilizar durante el proyecto (si es que el cliente no las aporta)
- Interviene en la propuesta del plan de trabajo
- Propuesta en la forma de venta del proyecto
- Evaluación a proyectos
- Evaluación a personal
- Atención al cliente
- Evaluación del cliente
- Asignación del personal al proyecto

#### **Área financiera**

- Venta y administración de los servicios de la consultora
- Propuesta comercial del proyecto

- Propuesta comercial de productos a entregar
- Propuesta de valuación de los recursos solicitados al proyecto
- Decisión del plan económico o comercial del proyecto
- Decisión el plan económico o comercial de los productos a entregar
- Decisión de la valuación de los recursos solicitados al proyecto
- Decisión de la forma de venta del proyecto
- Valuación de puestos de la consultora
- Evaluación a proyectos
- Evaluación a personal
- Atención al cliente
- Evaluación del cliente

Cabe mencionar que la dirección de una empresa consultora esta muy ligada a cada una de las áreas antes mencionadas, pues en si su trabajo principal es administrar a cada una de estas áreas, quizá al área que más se encuentra ligada es al área financiera, ya que esta administra los recursos financieros de la empresa.

En cierta forma podemos presentar al cliente como otra área mas dentro del proceso, sus funciones principales pudieran ser:

- Planteamiento y descripción del proyecto
- Planteamiento de las etapas del proyecto
- Planteamiento y decisión de los productos que recibirá
- Plan de recursos solicitados al proyecto
- Propuesta y decisión de tecnologías y herramientas a utilizar durante el proyecto
- Propuesta de un plan de trabajo
- Evaluación a proyectos
- Evaluación a personal
- Evaluación a la compañía consultora
- Supervisión del proyecto

- Propuesta en la forma de venta del proyecto
- Aceptación o rechazo del personal asignado al proyecto

Cabe mencionar, que hoy en día un gran número de consultorías cuentan con un departamento de calidad, dicho departamento como su nombre lo indica se encarga de supervisar que los servicios que ofrece la compañía consultora tengan calidad, entre mas competencia exista en el área de las empresas consultoras, es necesario que se cuente con mas calidad, es importante que cada proyecto que realiza la consultora cumpla con los requerimientos del cliente, y con los estándares manejados por las normas de la ISO-9000 para garantizar que se esta ofreciendo un buen producto al cliente.

### 3.2 OPCIONES DE SOLUCIÓN

Las empresas consultoras para dar una respuesta rápida y confiable al cliente, decide desarrollar un sistema para planificar y llevar el seguimiento de los proyectos.

Para llevar el control de la información recabada por el ejecutivo de cuenta y las áreas que involucra, la información debe estar en un repositorio, de tal forma que cualquier área involucrada la consulte, dependiendo de la acción que se desea realizar.

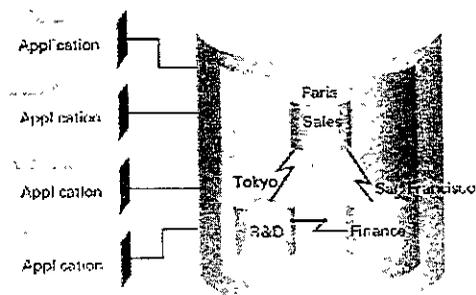


FIGURA 3.2.1 Llevar el control de la información

Este repositorio conocido como Base de Datos, debe contener la suficiente información para tener el control total de los proyectos que estén desarrollando y de esta forma, tomar la mejor decisión que así convenga.

Como se mencionó anteriormente, las empresas consultoras manejan demasiada información, de manera que el acceso a la Base de Datos debe ser rápida y confiable.

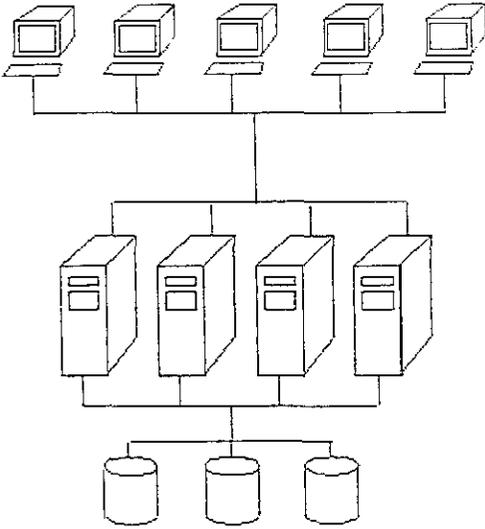
En el mercado existen varios productos de Bases de Datos. Los más importantes son:

- Oracle
- Informix
- Sybase
- SQL Anywhere
- SQL Server

Existen varios productos importantes para la elección de la herramienta de desarrollo para generar la interface del sistema.

- Visual Basic
- Visual C++
- Power Builder
- Visual Fox Pro

Estos productos deberán tener la capacidad de conectarse a las Bases de Datos más populares.



**FIGURA 3.2.2 Conectividad**

La elección de la Base de Datos y de la herramienta de desarrollo, dependerá principalmente de las funcionalidades de cada producto, además de contar con los requerimientos específicos de las empresas consultoras como son:

1. Ser producto líder en el mercado
2. Que los consultores tengan amplia experiencia en el uso del o los productos.
3. Fácil de aprender



**FIGURA 3.2.3 DBMS populares: Oracle, SQL Server, entre otros**

En este momento se han enunciado posibles opciones de solución.

Para poder tomar una decisión definitiva para la elección de los productos, describiremos cada uno de ellos para elegir la solución óptima.

### 3.2.1 Elección de la Solución Óptima

A continuación describiremos cada producto.

- Oracle

La distinción de Oracle como Base de Datos, es su presencia en casi todas las plataformas tales como SUN SOLARIS, HP UX, WIN 95/98/NT, AIX, DIGITAL, SEQUENCE, PYRAMID, AVIIION, LINUX, NCR, UNISYS, SCO, NOVELL, así como para los mas importantes protocolos de redes como: NETBIOS, TCP/IP, DECNET, todo esto a través de su producto de conectividad SLQNET (ahora NET8 Products)

Estas características lo hacen muy portable. Además de tener el 80% del mercado de Bases de Datos. Funciona en arquitectura cliente/servidor, ambiente multiusuario.

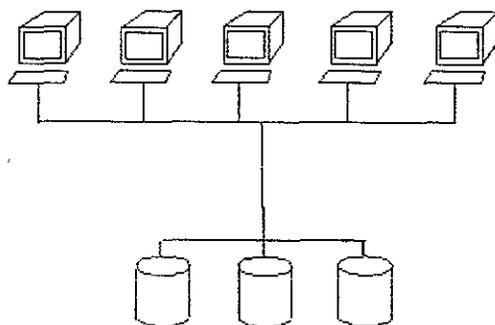


FIGURA 3.2.1.1 Ambiente cliente servidor, multiusuario

Oracle es utilizado como back-end por empresas muy importantes como: SAP, YAHOO, AMAZON.

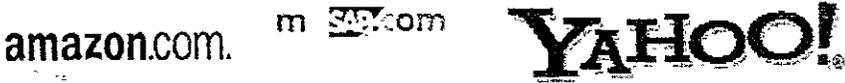


FIGURA 3.2.1.2 Yahoo, SAP, Amazon utilizan back-end ORACLE

Oracle esta diseñado para proveer todas las facilidades de almacenamiento y recuperación de información en diferentes formatos a través de sus utilerías o programas propias de Oracle como OEM (Oracle Enterprise Manager).

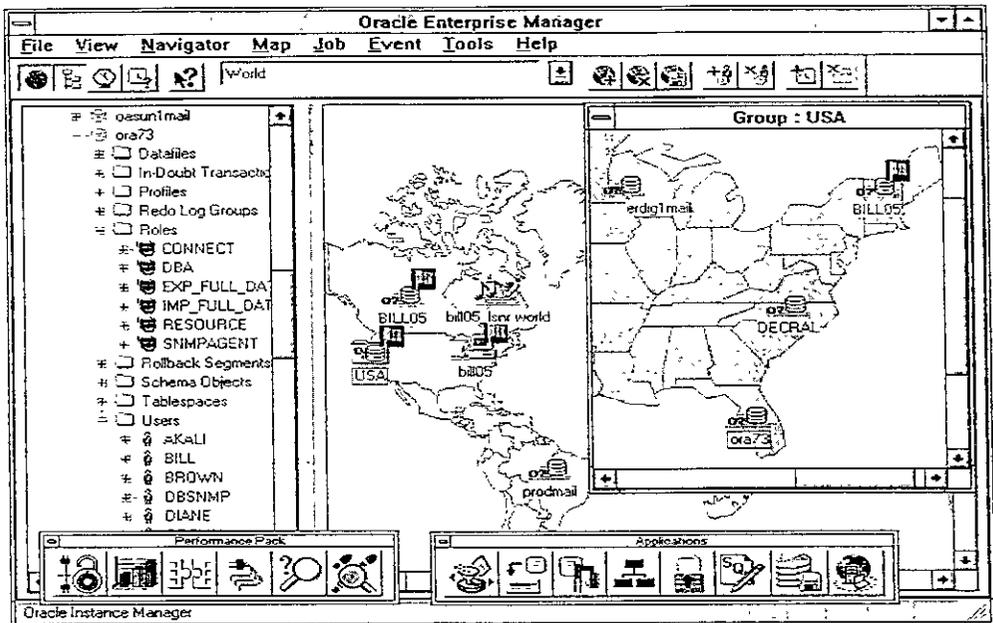


FIGURA 3.2.1.3 Aspecto de OEM, herramienta de administración de Bases de Datos Oracle

Maneja procedimientos almacenados y triggers. Cuenta además con opciones de seguridad a nivel sistema operativo y conexión remota.

Maneja los esquemas de disco espejo (Parallel Server).

Cuenta con bitácoras de transacciones, donde se almacenan las modificaciones que se realicen en la Base de Datos.

Para aumentar la velocidad de procesamiento y la consistencia de los datos, implementa el uso de grandes áreas de memoria SGA (System Global Area), para almacenar información.

Tiene desarrollados un gran número de procesos que elevan el desempeño de las rutinas de almacenamiento y recuperación de datos. Además de contar con analizadores de sentencias SQL y herramientas para la afinación de la Base de Datos

Subrutinas de almacenamiento y recuperación de datos aprovechan las ventajas de los sistemas que cuentan con varios procesadores.

Garantiza la integridad de los datos, llevando un control de los registros que han sido modificados dentro de una transacción a fin de que por algún motivo el proceso no se termine completamente (rollback).

- Informix

Proporciona alto desempeño y disponibilidad de información, incluye características como discos espejos (parallel server) y mecanismos de recuperación de información.

Trabaja bajo la plataforma cliente/servidor mediante el producto de conectividad Informix Net. Este soporta los protocolos TCP/IP y una variedad de computadoras y hardware de comunicación de datos como Ethernet, Token Ring.

The logo for Informix Software. The word "Informix" is in a bold, sans-serif font, with a stylized 'i' and 'x'. To the right of "Informix" is the word "SOFTWARE" in a smaller, all-caps, sans-serif font. A registered trademark symbol (®) is located between "Informix" and "SOFTWARE".

FIGURA 3.2.1.4. Informix

Cuenta con conjunto de tablas para registrar el diccionario de datos. En el caso particular del manejador de Bases de Datos On-Line se tiene la posibilidad de guardar en forma histórica las modificaciones hechas al diccionario de datos.

Soporta bases de datos distribuidas mediante el producto de conectividad Informix Star y el manejador de Base de Datos On-Line. Mediante estos productos se puede consultar y modificar varias bases de datos Informix en diferentes maquinas, como si fuera una sola.

Cuenta con recuperación de Bases de Datos de forma automática en casos de interrupciones por fallas de hardware o de energía. También se puede contar con una copia de las bases de datos en un disco diferente, a fin de que si el disco primario queda fuera de línea, el sistema se cambia al disco espejo en forma transparente para los usuarios.

Soporta el uso de Triggers y Procedimientos almacenados.

- Sybase

Sybase tiene una arquitectura multi-hilos, con su propio kernel que tiene integrado un monitor de control SQL.

Parte de Sybase, Transact SQL es un conjunto de extensiones para ANSI estándar de SQL es usado para escribir procedimientos almacenados.

Es posible construir triggers que se utilizan, entre otras cosas, para garantizar la importante propiedad de las bases de datos, denominada integridad referencial.



FIGURA 3.2.1.5 Sybase

El producto integra funciones de seguridad que permiten a los administradores de sistemas, controlar el personal que tendrá acceso a la base de datos y el nivel de autoridad con que lo hará.

Se utiliza un log de transacciones, en el cual se escribirá cualquier modificación de la que sea objeto la base de datos.

Como parte de su seguridad, Sybase proporciona la opción de generar discos espejos.

Cuenta con un optimizador de consultas, el cual analiza las sentencias SQL que el usuario introduce y si la consulta que el usuario realiza no es la optima, genera una nueva consulta que de manera más eficiente obtenga los datos requeridos.

- SQLAnywhere

FIGURA 3.2.1.6. Logo SQL Anywhere

A continuación se muestran los resultados de las pruebas que se hicieron para los manejadores de bases de datos (DBMS).

TIEMPO DE IMPORTACIÓN Y GENERACIÓN DE INDICES	Oracle	Informix	Sybase	SQL Anywhere
Bueno	X		X	X
Regular		X		
Malo				

Tabla 3.2.1.7 Tiempo de importación y generación de índices

<b>MECANISMOS DE SEGURIDAD</b>	<i>Oracle</i>	<i>Informix</i>	<i>Sybase</i>	<i>SQL Anywhere</i>
Bueno	X	X	X	X
Regular				
Malo				

Tabla 3.2.1.8 Mecanismos de seguridad

<b>MANEJO DE DIVERSOS TIPOS DE DATOS</b>	<i>Oracle</i>	<i>Informix</i>	<i>Sybase</i>	<i>SQL Anywhere</i>
Bueno	X	X	X	X
Regular				
Malo				

Tabla 3.2.1.9 Manejo de diversos tipos de datos

<b>PORTABILIDAD DIRECTA ENTRE PC'S</b>	<i>Oracle</i>	<i>Informix</i>	<i>Sybase</i>	<i>SQL Anywhere</i>
Bueno				X
Regular	X	X	X	
Malo				

Tabla 3.2.1.10 Portabilidad directa entre PC's

FACILIDAD DE INTERCONEXION ENTRE INFORMIX Y SYBASE	Oracle	Informix	Sybase	SQL Anywhere
Bueno			X	X
Regular	X	X		
Malo				

Tabla 3.2.1.11 Facilidad de interconexión entre Informix y Sybase

Para la calificación se estableció lo siguiente:

Bueno: 2

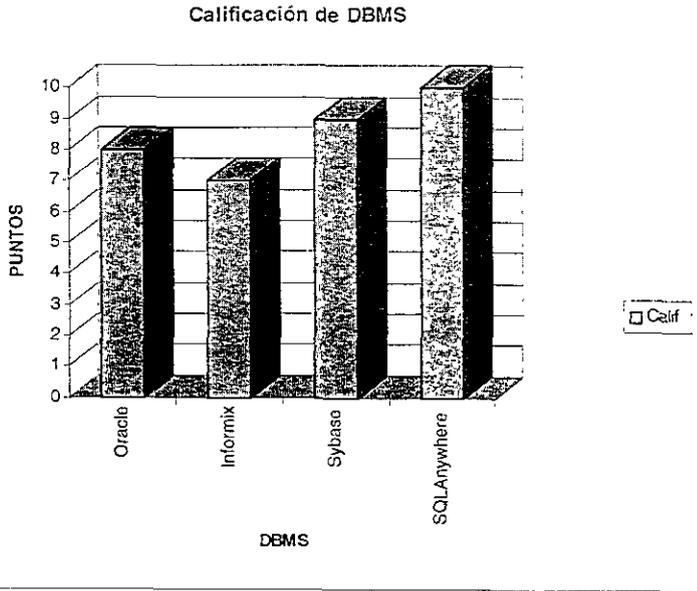
Regular: 1

Malo : 0

Tomando en cuenta las sumas totales tenemos la siguiente calificación para cada uno de los manejadores analizados:

DBMS	CALIFICACIÓN
Oracle	8
Informix	7
Sybase	9
SQL Anywhere	10

Tabla 3.2.1.12 Calificación de los DBMS



Gráfica 3.2.1.13 Calificación de los DBMS analizados.

### 3.2.2 Herramientas de Desarrollo

- Visual Basic

Este lenguaje está disponible para programadores de Windows. Visual Basic es una herramienta rápida para crear aplicaciones generales de negocios. Cuenta con una pequeña curva de aprendizaje, requiere pocos conocimientos de programación y tiene una rápida integración y rehusos de componentes.

Proporciona gran flexibilidad para llevar a cabo llamadas a archivos con extensión DLL, y agregar una gran cantidad de controles, al gusto del cliente.

Visual Basic constituye un camino fácil para crear interfaces de usuario, utilizando componentes preconstruídas, por lo que se vuelve un lenguaje flexible que cuenta con un editor-verificador de sintaxis y herramientas de perfeccionamiento (debugging).



FIGURA 3.2.2.1 Imagen Visual Basic

Es un sistema productivo para crear soluciones en ambiente Windows, integra controles visuales preconstruidos por terceros.

Maneja multimedia y a través de ODBC, accesa a varias Bases de Datos. Permite un ensamble fácil y rápido entre la interface de un usuario y los componentes prefabricados.

Permite manipular otras aplicaciones para utilizarlas como componentes en aplicaciones propias (Word, Excel, etc.) siempre y cuando dichas aplicaciones soporten OLE.

La desventaja de Visual Basic es que su código no es compatible en otras plataformas que no sean Windows.

Se auxilia de lenguajes más robustos, como "C", Pascal, para hacer manipulación de memoria, accesos directos al sistema operativo.

- Visual C++

Es una herramienta poderosa y flexible, para generar soluciones de negocios, además de ser una herramienta rápida para programadores que utilizan C y C++.

Incluye una gran variedad de librerías preconstruidas, controles visuales.

Permite la reutilización total del código.

Maneja controles VBX (Visual Basic eXtension), los cuales son componentes preconstruidos que aceleran el desarrollo de software compatibles con Visual Basic.



FIGURA 3.2.2.2 Logo Visual C

- Power Builder

Permite el desarrollo de aplicaciones robustas, bajo ambientes multiplataformas, cuenta con un optimizador de código, posibilidad de distribución de objetos en un ambiente de red y manejadores nativos para diferentes bases de datos.

Posee un soporte completo para ambiente Windows.

Posee una familia de herramientas de desarrollo escalable que incrementan la productividad de las aplicaciones. Soporta desarrollo en grupo

La migración entre servidores SQL es sencilla.

Se pueden crear bases de datos desde Power Builder.



FIGURA 3.2.2.3 Logo Power Builder

Basándonos en las características se deben seleccionar las herramientas de desarrollo adecuadas, así como el manejador de la base de datos para desarrollar el sistema.

Para la calificación de las herramientas de desarrollos, se tomaron lo siguientes criterios:

<b>CONECTIVIDAD CON SQL ANYWHERE</b>	<i>Visual Basic</i>	<i>Visual C++</i>	<i>Power Builder</i>
Bueno			X
Regular	X	X	
Malo			

Tabla 3.2.2.4 Facilidad de conectividad con Sybase/SQL Anywhere

<b>HERRAMIENTAS MEJORADAS PARA SQL ANYWHERE</b>	<i>Visual Basic</i>	<i>Visual C++</i>	<i>Power Builder</i>
Bueno			X
Regular	X	X	
Malo			

Tabla 3.2.2.5 Herramientas mejoradas para SQL Anywhere

<b>DEFINICIÓN, CORRECCION, COMPILACIÓN DE C++</b>	<i>Visual Basic</i>	<i>Visual C++</i>	<i>Power Builder</i>
Bueno		X	X
Regular	X		
Malo			

Tabla 3.2.2.6 Definición, compilación y corrección de clases integradas de C++

MÉTODO DE CONEXIÓN A SERVIDOR SQL	<i>Visual Basic</i>	<i>Visual C++</i>	<i>Power Builder</i>
Bueno	X	X	X
Regular			
Malo			

Tabla 3.2.2.7 Método de conexión a servidores SQL a través de ODBC

DESARROLLO DE APLICACIONES CONTROLANDO EL BACK-END	<i>Visual Basic</i>	<i>Visual C++</i>	<i>Power Builder</i>
Bueno			X
Regular			
Malo	X	X	

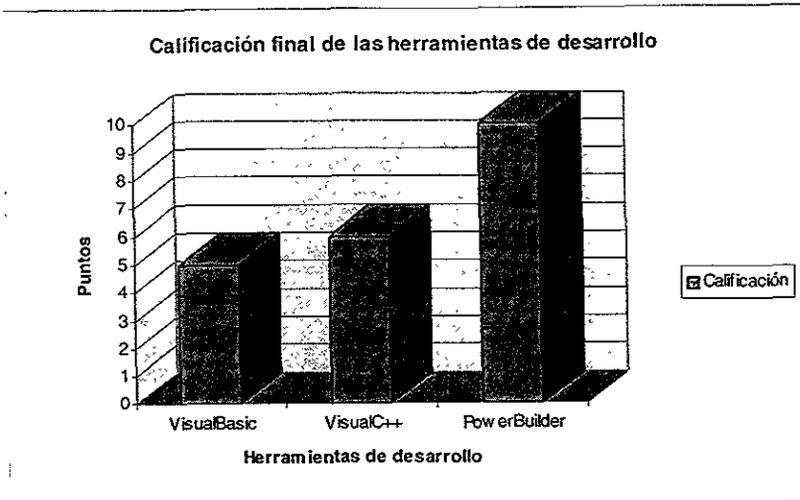
Tabla 3.2.2.8 Desarrollo de aplicaciones controlando el back-end (SQL Anywhere)

Tomando la escala anterior de los DBMS, tenemos:

HERRAMIENTA DE DESARROLLO	CALIFICACIÓN
Visual Basic	5
Visual C++	6
Power Builder	10

Tabla 3.2.2.9 Calificación final

Mostramos en forma gráfica la calificación final de las herramientas de Desarrollo analizados.



**Gráfica 3.2.2.10 Calificación final.**

Como podemos observar, la mejor integración con el DBMS y la herramienta de desarrollo son SQL Anywhere y Power Builder.